NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MODELING OF OPERATING TEMPERATURE PERFORMANCE OF TRIPLE JUNCTION SOLAR CELLS USING SILVACO'S ATLAS

by

Michael H. Sanders

September 2007

Thesis Advisor:	Sherif Michael
Second Reader:	Todd Weatherford

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 2007	Master's Thesis

4. TITLE AND SUBTITLE Modeling of Operating Temperature Performance of Triple Junction Solar Cells Using Silvaco's ATLAS	5. FUNDING NUMBERS
6. AUTHOR(S) Michael H. Sanders	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited	

13. ABSTRACT (maximum 200 words)

The inefficiency of solar cells due to high operating temperatures presents a growing issue for the spacecraft industry. Currently, the problem is solved by accepting the manufactured designs and compensating for losses with larger solar arrays. Building upon prior thesis work at the Naval Postgraduate School, this thesis utilizes Silvaco's ATLAS software as a tool to simulate the performance of a typical InGaP/GaAs/Ge multi-junction solar cell at various temperatures. Additional optimization is performed on the base thickness layers to represent that enhancement for the proper operating environment can be achieved. Results are shown for a multi-junction cell operating under Air Mass 0 at 300K, 325K, 350K, and 375K.

14. SUBJECT TERMS Silvaco, Atlas, Multi-junction Solar Cell, Temperature, Photovoltaic, Simulation	15. NUMBER OF PAGES 123
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MODELING OF OPERATING TEMPERATURE PERFORMANCE OF TRIPLE JUNCTION SOLAR CELLS USING SILVACO'S ATLAS

Michael H. Sanders
Lieutenant Commander, United States Navy
B.S., Carnegie Mellon University, 1996

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 2007

Author: Michael H. Sanders

Approved by: Sherif Michael
Thesis Advisor

Todd Weatherford
Second Reader

Jeffrey B. Knorr
Chairman, Department of Electrical and
Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The inefficiency of solar cells due to high operating temperatures presents a growing issue for the spacecraft industry. Currently, the problem is solved by accepting the manufactured designs and compensating for losses with larger solar arrays. Building upon prior thesis work at the Naval Postgraduate School, this thesis utilizes Silvaco's ATLAS software as a tool to simulate the performance of a typical InGaP/GaAs/Ge multi-junction solar cell at various temperatures. Additional optimization is performed on the base thickness layers to represent that enhancement for the proper operating environment can be achieved. Results are shown for a multi-junction cell operating under Air Mass 0 at 300K, 325K, 350K, and 375K.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank Professors Michael and Weatherford for their assistance and guidance during this time. I would also like to thank Professor Haegel for her time and explanations of optics that gave insight into one of the many issues faced. As for understanding Silvaco and MATLAB code, I would like to thank Dr. Jones and Eric Jackson, respectively. Lastly, as with all my endeavors, I would like to thank my wife and family for their patience and encouragement.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The space industry is continually investigating for a new means to supply its energy hungry satellites with more power. Solar cells have been supplying this needed energy since the first launches into orbit. The latest technologies use exotic materials in various layers to utilize more of the sun's electromagnetic spectrum to convert more light energy into electrical energy. With so many layers and materials, the prediction of the performance of the cells has grown into an enormous predicament for designers; how to predict performance parameters for a complete multi-junction solar cell taking into account such a large number of interrelated variables while still taking into account the operating environment? The current method of fabricating a cell, then testing it with Xenon light to represent sunlight, is costly and prohibitive due to the number of variable factors in a multi-junction solar cell. The size of this question is well beyond the scope of this thesis but it addresses part of the question. As more data is available from actual satellite systems, the impact of solar cells not operating at the designed temperature of 28°C comes into question. More importantly, are solar cells designed for optimal performance at these higher temperatures?

Research at the Naval Postgraduate School has shown a simulation tool that can address this question. The ATLAS™ tool in Silvaco Virtual Wafer Fabrication software has been shown to predict various solar cells with different effects simulated. Previous thesis work has demonstrated the feasibility of accurately simulating single, dual, and triple-junction solar cells under space sunlight (Air Mass 0) conditions. A multi-junction cell can be closely simulated even though the tunnel junction that is between each cell is treated as a vacuum for this paper. Silvaco ATLAS™ has also been used to simulate radiation effects on cells and to simulate performance based on different input spectra. This thesis uses ATLAS™ to alter the temperature of a triple-junction solar cell to

incorporate the effects of different operating temperatures of spacecraft and to predict if a more optimal cell could be manufactured to take into account the different operating temperatures.

A key component was to verify the previous works and establish that baseline in which to compare results. Due to the limited time and computer programming skills, this thesis addresses only two variables for each test. First, a single junction cell was used to observe the effects of changing temperature. A GaAs single junction cell was utilized to observe the decrease in voltage as the temperature increased (see Figure 1).

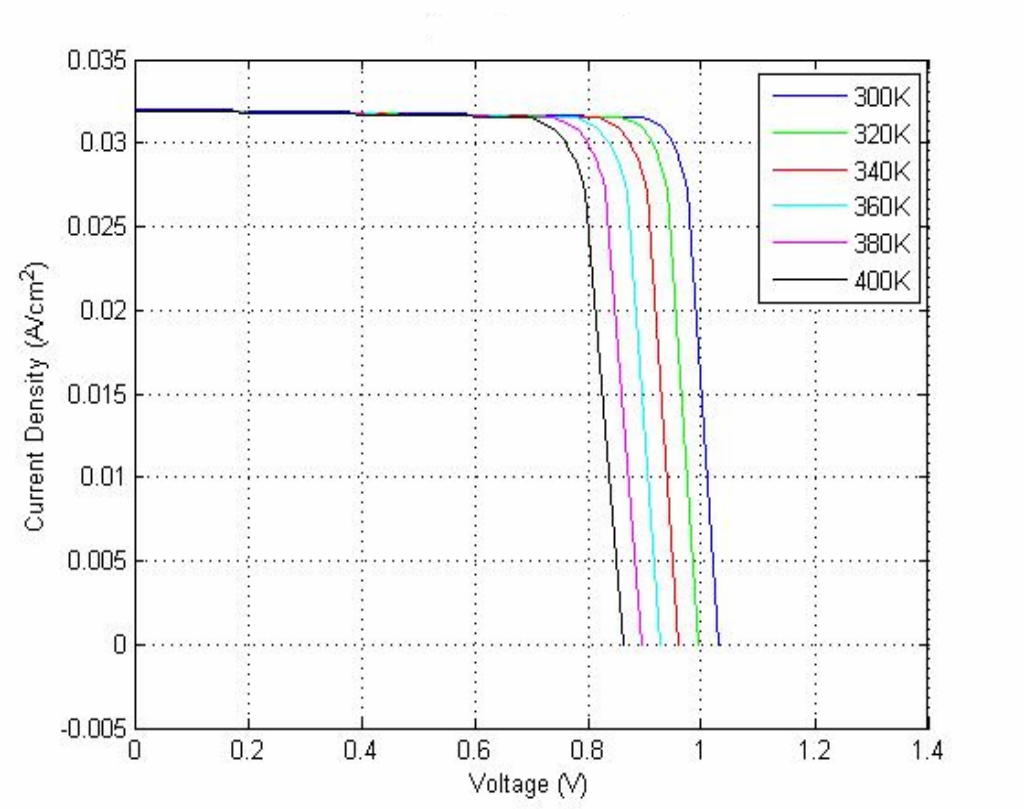

Figure 1. Simulated GaAs single junction cell showing temperature variation.

Next, a triple-junction cell programmed by Andrew Bates was selected as the test cell. A MATLAB program file originally written by Bradley Davenport, later modified by Burt Canfield, was used to vary two cell parameters at a time to note

a change in maximum power and efficiency. While this small variation would never allow the optimization of the cell, it would illustrate a trend of the effects of changing thickness and doping levels.

An inherited error has been the lack of a tunnel junction in the simulation cell. Simulated numbers have matched closely to posted cell specifications, but a true multi-junction cell has effects due to the tunnel junction that are not being simulated. A graduate of NPS, Robert Gelinas, was not able to properly simulate the tunnel junction. At this time, Aerospace Corporation has been working on this issue and may have a solution in the near future.

Continued research using Silvaco software to simulate solar cell design will improve the design process, permitting more manufacturing and environmental factors to be considered. The objectives of this research were to: Select a suitable multi-junction solar cell,

- Verify the simulation both at room temperature and at higher temperatures for single and multi-junction cells,
- Vary cells parameters, mainly thickness and doping, to note any improvements in the cells efficiency,
- Demonstrate that Silvaco can be a design tool for solar cells.

This thesis uses the software to note how changes in the environment can potentially influence the cells operations and to investigate way to find an optimum cell designed for the operating temperature.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

In 1839, physicist Alexander-Edmond Becquerel discovered the photovoltaic cell that converts light energy to electrical energy [1]. It was not until 1883 that the first solar cell was built, using selenium to form a cell that was about 1% efficient [1]. In 1946 Russell Ohl patented the modern solar cell, ushering the age of solar power technology [1]. Bell Laboratories continued to improve the cell to about 6% efficiency [1]. Modern cell design has transitioned to much higher efficiencies.

First-generation solar cells, consisting of a single layer p-n junction, have limited power production due to only being able to utilize a portion of the input spectrum. Second-generation materials are based on the use of "thin-film" deposits of semiconductors, such as amorphous silicon or cadmium telluride. Even though these second-generation thin film cells are typically less efficient than first generation silicon cells, the lower costs in manufacturing has achieved a lower cost per watt [1]. Multi-junction cells use different materials to vary band gap energies to utilize more of the spectrum, thus producing more electrical output. The highest triple junction cells in production are around 28% efficient with Spectrolabs reporting at the 2007 Aerospace Corporation Space Power Workshop tests for a 30% efficient cell by 2008 [2].

As great improvements in solar cell design continue to stretch the limits of the materials and designs, the virtual modeling of solar cells is necessary to continue to improve its power and efficiency.

B. OBJECTIVE AND APPROACH

The objective of this thesis was to compare the efficiency of a triple junction cell at room temperature to more realistic operating temperatures, and

further, to determine if physical variations of the cell's parameters could produce a cell with higher efficiency. The approach started with single and triple junction baseline cells based on Bates' [3] previous work. Once both cells could be simulated in ATLAS, temperature variations were introduced. First, a single junction GaAs cell was simulated to note the decrease in voltage as the temperature increased. Next, a triple junction cell was used to note the temperature variance. After the temperature dependence of a triple junction cell was demonstrated, the thicknesses for the bases of the top two cells were varied as temperatures changed. Next, the doping concentrations for the bases to the top two cells were varied as temperature changed. Due to time and programming limitations, only two cell parameters were changed for each change in temperature. Finally, the best cells were selected and critiqued for improvements in efficiency and power. For all efficiency calculations, P_{in} was 135.3 mW/cm^2.

C. RELATED WORK

The Naval Postgraduate School has had many years in research on using Silvaco to simulate a solar cell, starting in 1999 with Darin McCloy [4]. McCloy presented the first modeling of a high efficiency solar cell in Silvaco. The next big step was done by Michalopoulos [5]. He attempted to further optimize a triple junction cell by varying the thicknesses of the individual junctions. Green [6] followed Michalopoulos' work by simulating a quad-junction solar cell. Realizing the complexity and the number of possible variables the make a multi-junction solar cell, Bates [3] developed an algorithm to further optimize Michalopoulos and Green's work. Bates also demonstrated that a solar cell can be optimized based on environmental factors, specifically an optimal cell for the Martian light spectrum. Canfield [7] demonstrated the effects of temperature variation on thermopholtovoltaic (TPV) devices. This thesis draws upon the original work of Michalopoulos, taking into account the demonstrated effects of the environment

by Bates. Finally, since a temperature variation was the environmental factor selected, Canfield's work with TPV had a direct influence on this thesis.

Outside of NPS, research has mainly focused on developing more efficient solar cells by utilizing more of the Air Mass Zero (AM0) solar spectrum. Thus multi-junction solar cells were developed. A single junction cell only uses part of the light spectrum, such as wavelengths from 0.6 to 0.9 μm for a GaAs solar cell, while a multi-junction cell uses different materials to more fully exploit the entire spectrum (see Figure 2). Some research has demonstrated the dependency of efficiency on temperature, such as Linder and Hanley in [8]]. Figure 3 is taken from their results. Their paper states that triple junction cells degrade more quickly due to temperature increases, but it does not attempt to optimize any cells for temperature changes by altering the cell parameters [8].

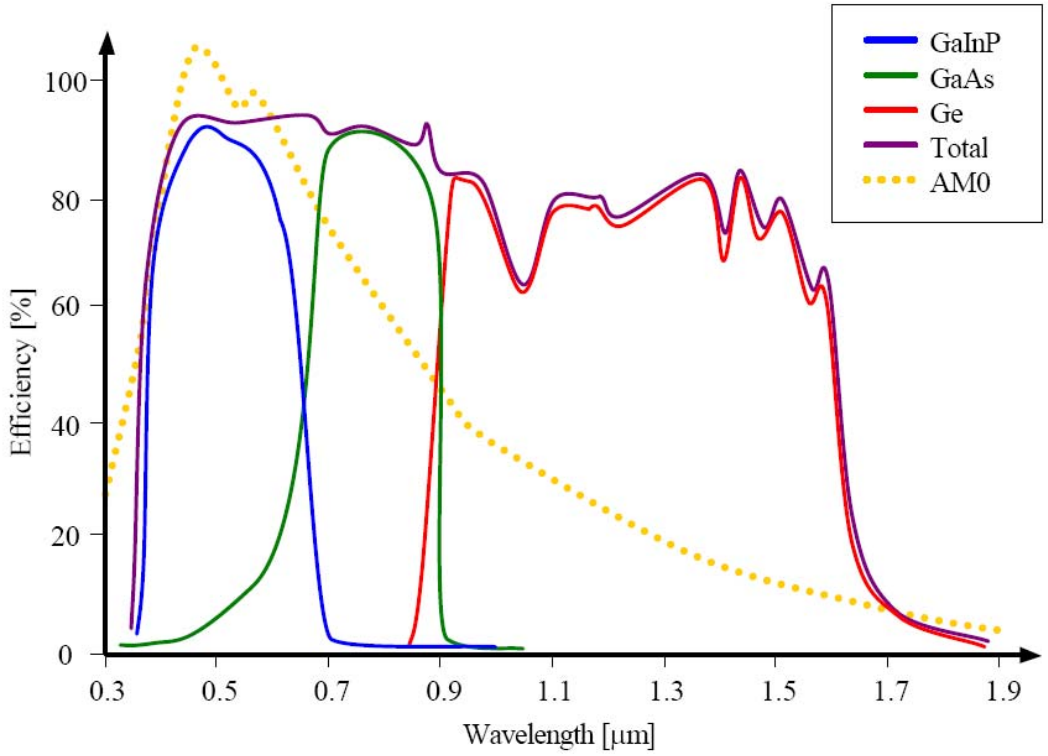

Figure 2. Absorption efficiency of the different materials of a triple junction solar cell [from 5]

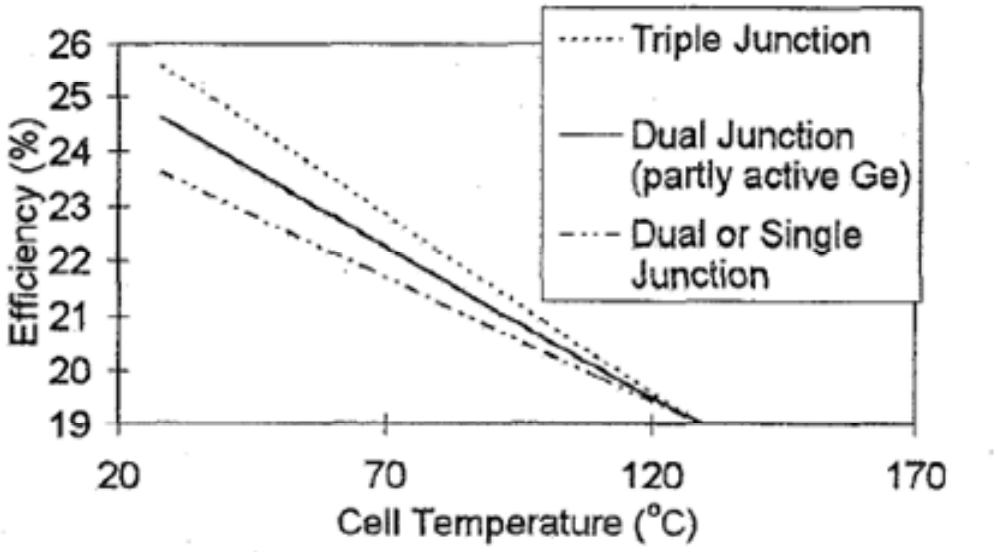

Figure 3. Temperature response for single, dual and triple junction solar cells [from 8]

D. THESIS ORGANIZATION

A basic semiconductor physics background is presented for completeness of the subject. Solar cell properties are investigated, followed by a discussion of the effects of temperature on solar cells. Next, the simulation software is presented in general terms for understanding with specific highlights about this particular work. Finally, the results of temperature variations on triple junction cells are presented with concluding remarks.

II. PHOTOVOLTAICS

A. SEMICONDUCTOR PHYSICS

As the name implies, semiconductors are materials that have physical properties between conductors and insulators. This middle property has made the modern computer age possible. Photovoltaic materials exploit these properties to generate electricity from photons.

1. Basic Semiconductor Models

a. Bohr Model

The Bohr model represents the structure of an atom, showing the relationship of the nucleus of protons and neutrons surrounded by orbiting electrons. Three assumptions help to define Bohr's model.

- The electrons have certain stable, circular orbits about the nucleus.
- The electrons can shift to other orbits by gaining or losing energy equal to the difference in energy levels (see Figure 2).
- The angular momentum of the electron is always and integral multiple of Planck's constant divided by 2π, represented as $\hbar$ [9]

This angular momentum was coupled directly to energy. Assuming the electron's angular momentum is an integer times then the electron binding energy, $\mathrm{E_h}$, for that orbit or shell can be solved by:

$$E_h = -\frac{m_0 q^4}{2(4\pi\varepsilon_0 \hbar n)^2}, n = 1, 2, 3, \cdots$$

Where m_0 is the mass of a free electron, q is the magnitude of the electronic charge, ε_0 is the permittivity of free space, and n is the orbit identifier [9]. The quantized energy transition from a higher to lower n explains why only certain

wavelengths of light are emitted. An easy way to think of idea of only discreet orbits is to assume that the electron is a wave and only an integral number of wavelengths can fill a particular orbit. Otherwise, the electron would create self-interference and be able to maintain the orbit [10]. The importance of the Bohr model is that the energy of electrons is restricted to a limited set of values.

b. Bonding Model

Going from the single atom of the Bohr model to a physical structure, the bonding model takes into account the interaction of the weakly bonded, outer shell electrons called valence electrons. Group *I-V* elements (see Figure 4 [11] for periodic table of elements), such as Si, have four valence electrons that can be shared among other atoms to develop a lattice structure. Missing atoms or defects can be represented with the bonding model as well as the freeing of an electron from its atom to atom bond (see Figure 5). The Bonding Model helps to visualize the spatial aspects of the electrons but does not adequately address the energy aspects. The Energy Band Model addresses the various energy levels of the electrons.

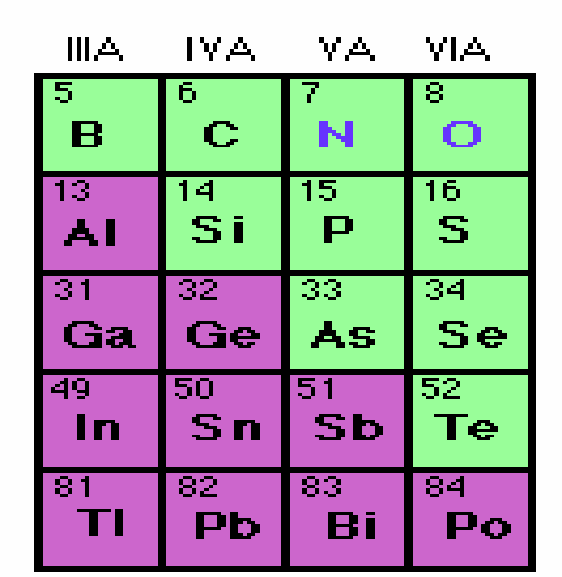

Figure 4. Portion of periodic table of elements [after 11]

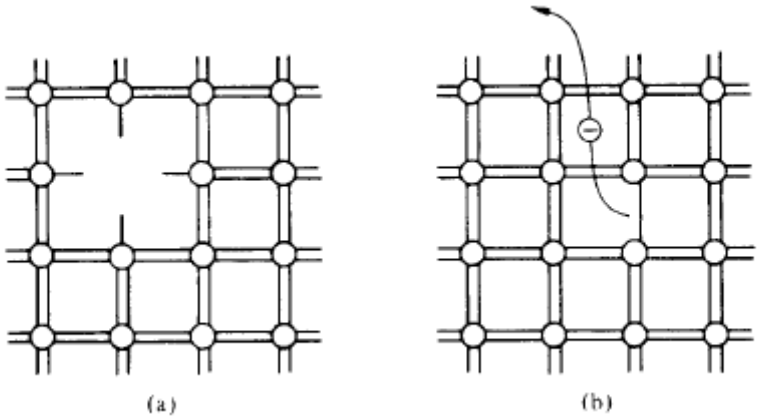

Figure 5. Using the bond model, visualization of (a) defect or missing atom and (b) an electron breaking its bond from an atom [from 12].

c. Energy Band Model

Relating back to Bohr's model and the discreet energy levels of electrons, the spread in allowed energy states leads to a set of energy bands. At the atomic level, the allowed states consist of two bands separated by an intervening energy gap. The valence band represents the lower band of allowed states. The conduction band represents the higher energy band state [12]. In between these two states is the band gap (see Figure 6).

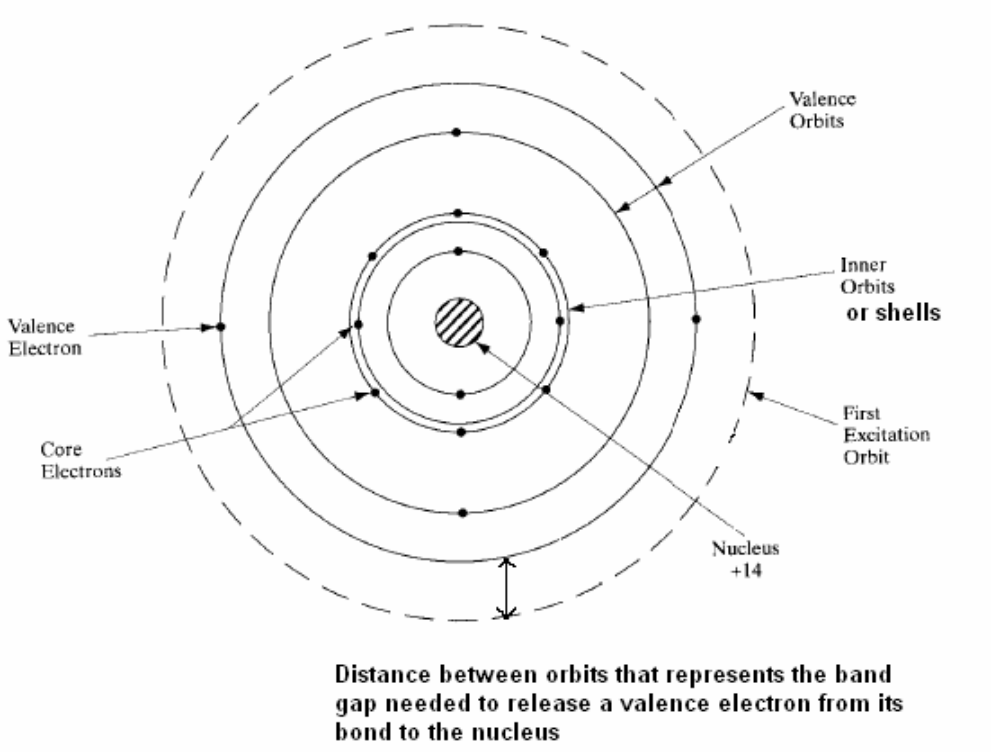

Figure 6. An orbital model of a Si atom showing the electrons orbiting the nucleus and the first excitation orbit that would free an electron from its bond [after 9].

With the models established, the question of current flow can be introduced. Current is the movement of charged carriers. For our purposes, carriers can be either electrons for negatively charged particles or holes, which are actually the absence of electrons within a lattice structure, for positively charged particles. The conduction band uses electrons as its carriers while the valence band uses holes as its carriers. When an electron receives enough energy to move from the valence band to the conduction band, it has broken its atomic bond and is now free to move throughout the material (see Figure 7).

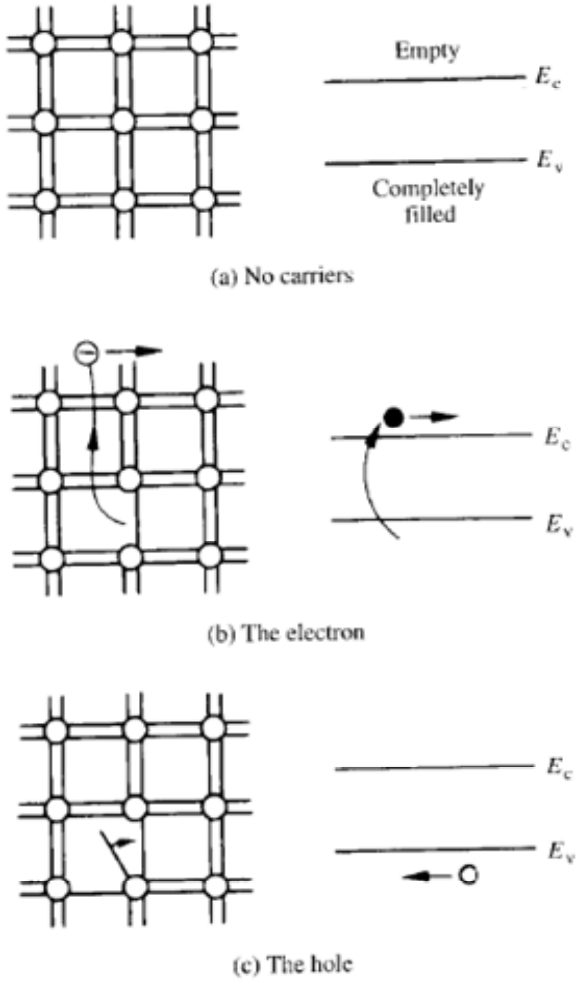

Figure 7. Using the bonding model (left) and the energy band model (right), a visualization of (a) no carrier situation, (b) electron carrier, and (c) hole carriers [from 12].

In this transition of energy states, normally a hole is formed in the valence band, left by the vacancy of the electron. This exchange is termed *electron-hole pair generation*. The band gap and the generation of carriers determine the electrical properties of the material.

2. Material Properties

Basically, there are three electrical material types; insulators, conductors, and semiconductors. In band gap terms, an insulator has a very wide band gap, thus very few carriers exist inside the material and is a poor conductor of carriers.

A conductor has a very narrow or overlapping band gap, thus an abundance of carriers are present making it an excellent conductor of carriers. Metals are great conductors due to their overlapping bands. As its name implies, semiconductors have a band gap energy in between conductors and insulators. In this thesis, by exciting the electrons in the valence band to the conduction band with photon stimulation, creates carriers for conduction. See Figure 8 for a representation of the differences in materials. The conduction of semiconductors can be manipulated by the addition of impurities into the lattice structure, called doping.

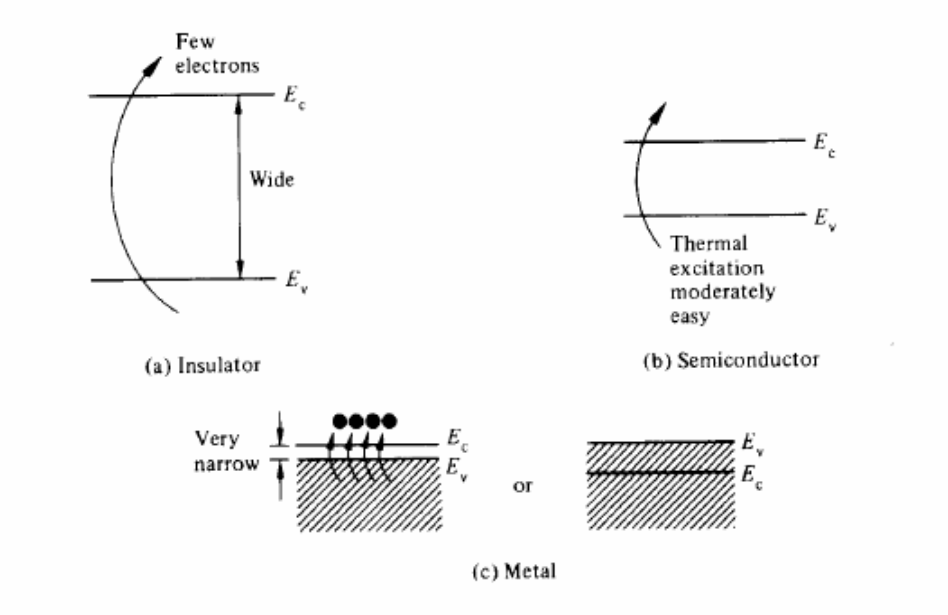

Figure 8. Using the band gap model, (a) an insulator, (b) a semiconductor, and (c) a conductor or metal [after 12].

3. Doping

Doping is the addition of controlled amounts of specific elements with the purpose to increasing either the electron or hole concentrations [12]. A material that is pure without a dopant is classified as an intrinsic material. The addition of a dopant creates an extrinsic or doped material. Taking a look at the Periodic Table (see Figure 4 above), common dopants are in the Group III and V

elements, such as B, Ga, or In; or P, As, or Sb respectively. Hole-increasing dopants are called acceptors and are part of the Group III column. Electron-increasing dopants are called donors and are in the Group V column. Using the bonding model, a donor has one extra electron after the establishment of its inter-atomic bonds. Likewise, an acceptor generates a hole in the lattice due to its lack of an electron for inter-atomic bonding (see Figure 9 [12]). At thermal equilibrium, the concentration of majority carriers is a constant, $p \times n = n_i^2$, where p is the holes and n is the electrons and n_i is the undoped number of majority carriers. Introducing a dopant into a semiconductor creates different properties for the material.

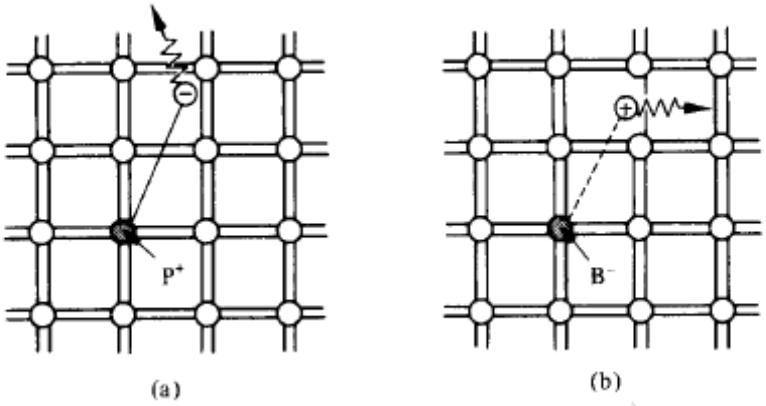

Figure 9. Bonding model illustration of (a) donor, P, contributing an electron to the lattice and (b) acceptor, B, accepting an electron from the lattice forming a hole [from 12]

There are two types of doped materials that can result. An n-type material results from a donor dopant, creating excessive numbers of electrons, while a p-type material is created from excessive numbers of holes resulting from an acceptor dopant. The excessive carriers are the majority carriers and can be thought of as analogous to the type of doping. The effect of doping can greatly alter the electrical properties of a material. Figure 10 uses a band diagram to illustrate how a semiconductor can have more electron or hole conduction based on the doping properties.

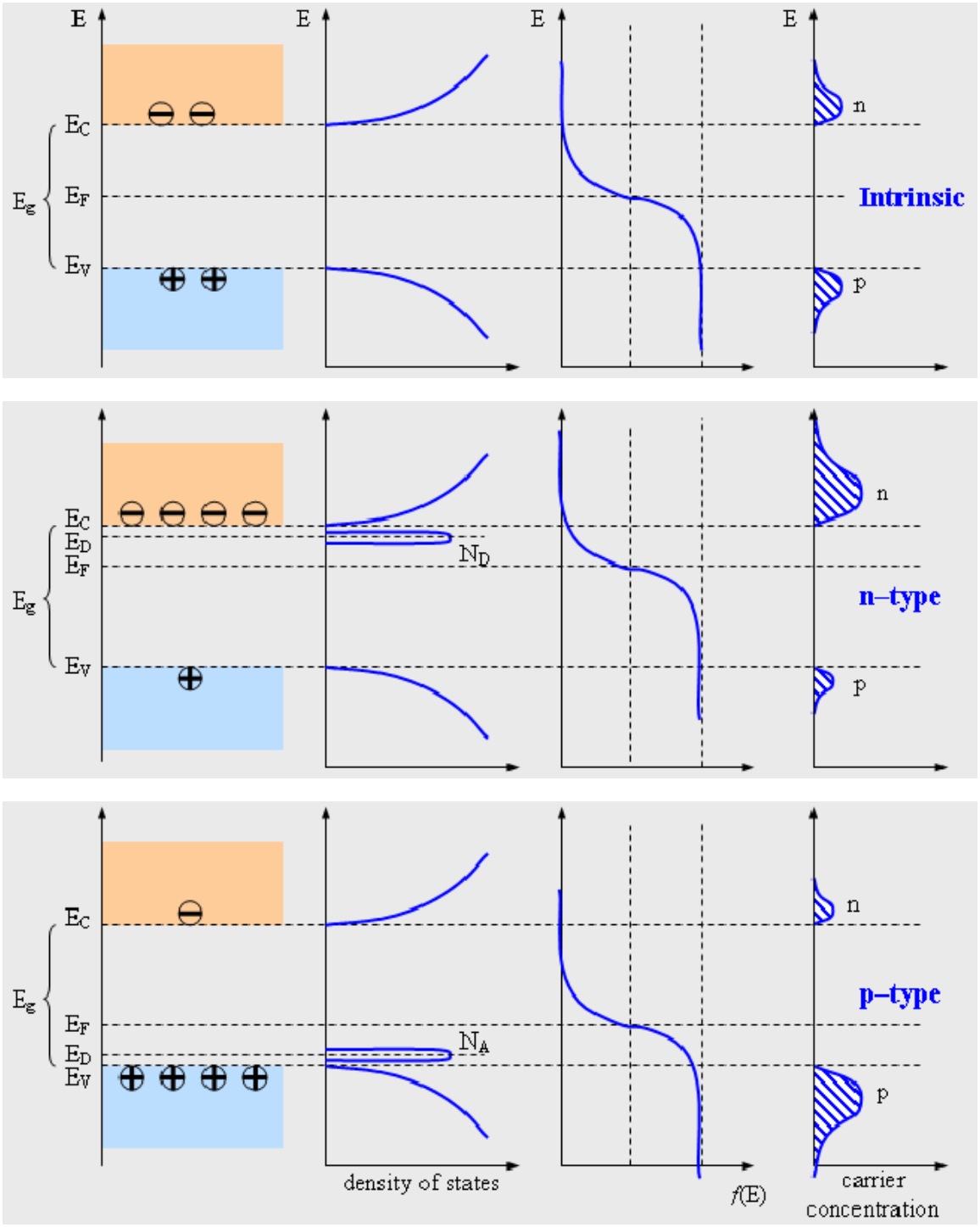

Figure 10. Band diagram, density of states, Fermi function, and carrier concentrations for (top) intrinsic material, (middle) n-type material, and (bottom) p-type material [from 5]

4. P-N Junction

For solar cells the interaction between the p-type and n-type material, when joined, creates a p-n junction. The p-n junction is the basis for modern electronics and creates the condition for electronic power generation from light. The excessive majority carriers diffuse across the boundary to the other material (see Figure 11), creating an area that is devoid of majority carriers. As diffusion continues, this area develops an electrical potential that will eventually prevent carriers from crossing the boundary. The space around the boundary is called the depletion region.

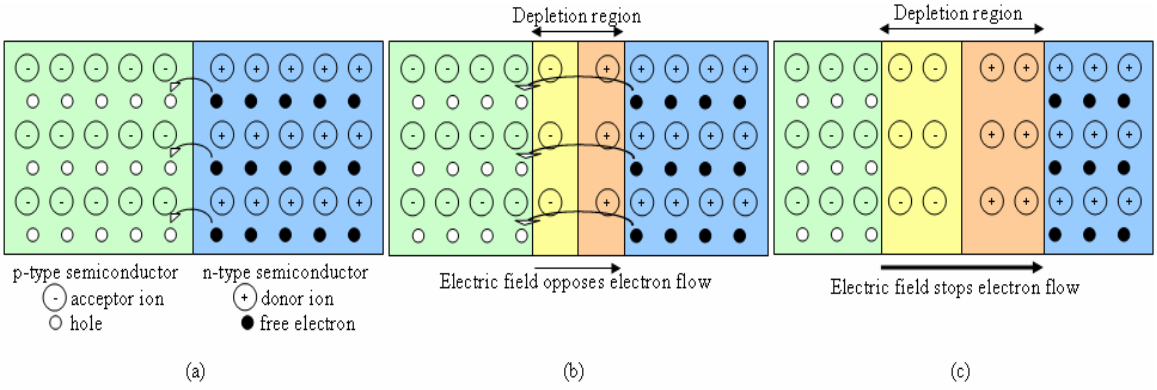

Figure 11. Formation of the depletion region at a p-n junction (electric field arrows are in terms of electron flow) [from 3].

Using a different approach, Figure 12 displays the band energy levels for a p-n junction and equilibrium. The electrostatic potential is analogous to the depletion region and illustrate the reason the electron-hole pair separate due to the potential difference over this region.

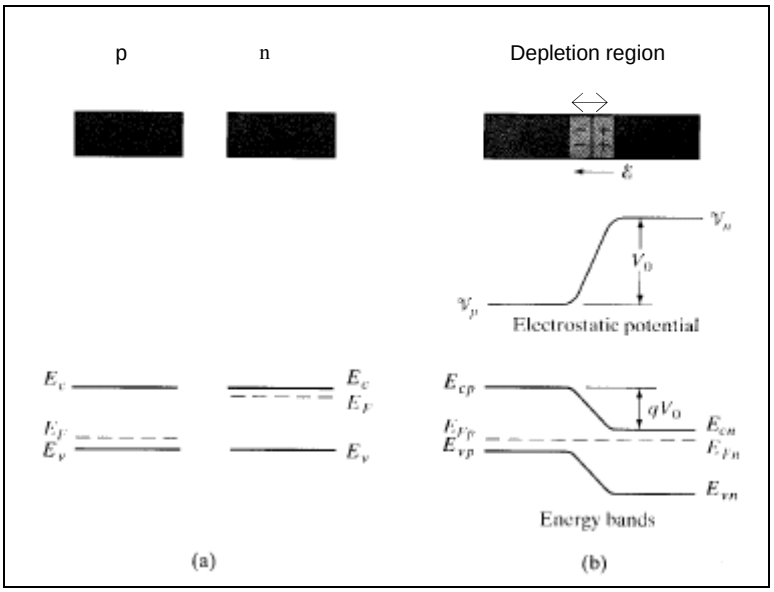

Figure 12. Properties of an equilibrium p-n junction showing (a) isolated, neutral regions and (b) junction showing depletion region, the resulting electrostatic potential, and the energy bands [after 9]

An important note for solar cells, the electric potential also acts in separating electron-hole pairs that are generated, as explained in the next section [13].

B. SOLAR CELLS

1. Solar Cell Operations

The foundation of all solar cells is the basic p-n junction, or a simple diode. A diode used in an electrical circuit is covered with an opaque insulation in order to prevent light from interfering with its operations. A solar cell is a diode that utilizes incident light to generate an electrical current.

To generate current, a photon collides with a valence electron, imparting energy to the electron. If the electron gains energy equal to or greater than the band gap energy of the material then the electron will be freed from the weak, valence bond. Thus, the freed electron creates an electron-hole pair. This process is called photogeneration. The electrical potential in the depletion region sweeps the electron toward the n material and hole toward the p material [13]. If

the electron can reach the n-side of the material without recombining, electrical current forms that can power an externally connected circuit (see Figure 13).

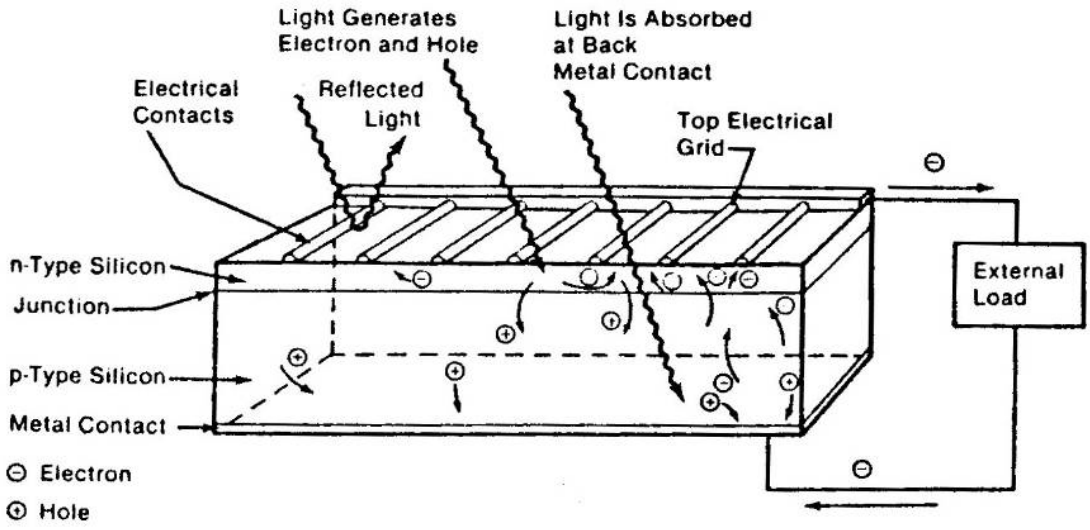

Figure 13. Photogeneration in a simple solar cell [from 13].

2. Solar Cell Performance

Many different factors affect solar cell performance. Doping concentrations, types of materials, thickness of layers, lattice growth, and even the manufacturing process can influence a cells performance. Comparing different solar cells can be difficult so performance comparisons use uniform measurements and parameters to compare.

If the cell is exposed to light but unconnected to an external circuit, a voltage is developed that opposes any further carrier diffusion. This voltage at this equilibrium point is the open-circuit voltage (V_{oc}) and is the maximum voltage that the cell can produce [14].

Likewise under short circuit conditions, the electrons flow to the metal contacts without a potential build-up, a short-circuit current (I_{sc}) materializes. This current is the maximum current that the solar cell can supply [14]. The

number of electron-hole pairs depends on the intensity and wavelength of the incoming light, which in turn determines the amount of current produced [13].

The characteristics of a solar cell are best illustrated using a current-voltage curve (called an *I-V* curve). As can be seen in Figure 14, the voltage is the horizontal axis and the current is the vertical axis.

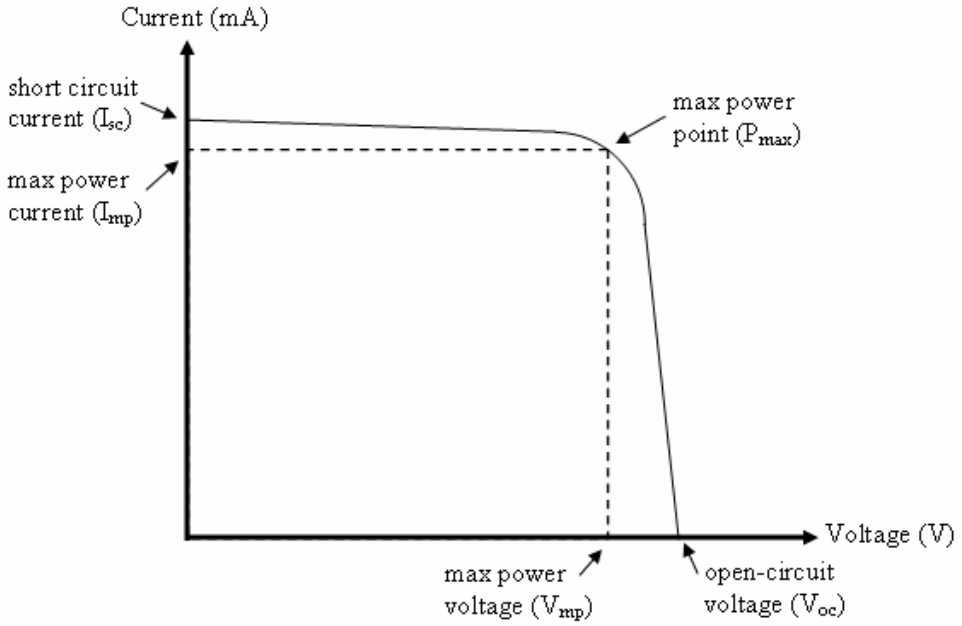

Figure 14. Typical current-voltage relationship for a simple solar cell [from 3].

At zero voltage, the cell can produce the most current. As the external load increases, the voltage will increase as the current decreases until V_{oc} is reached. At $V_{oc,}$ all excess carriers have recombined within the cell and no current is available to power the load [13]

Two other parameters are used in this thesis, maximum power (P_{max}) and efficiency (abbreviated eff or η). P_{max} is the point of at which maximum power is produced by the solar cell (see Figure 15). V_{mp} and I_{mp} can then be determined for use in the design of a solar array.

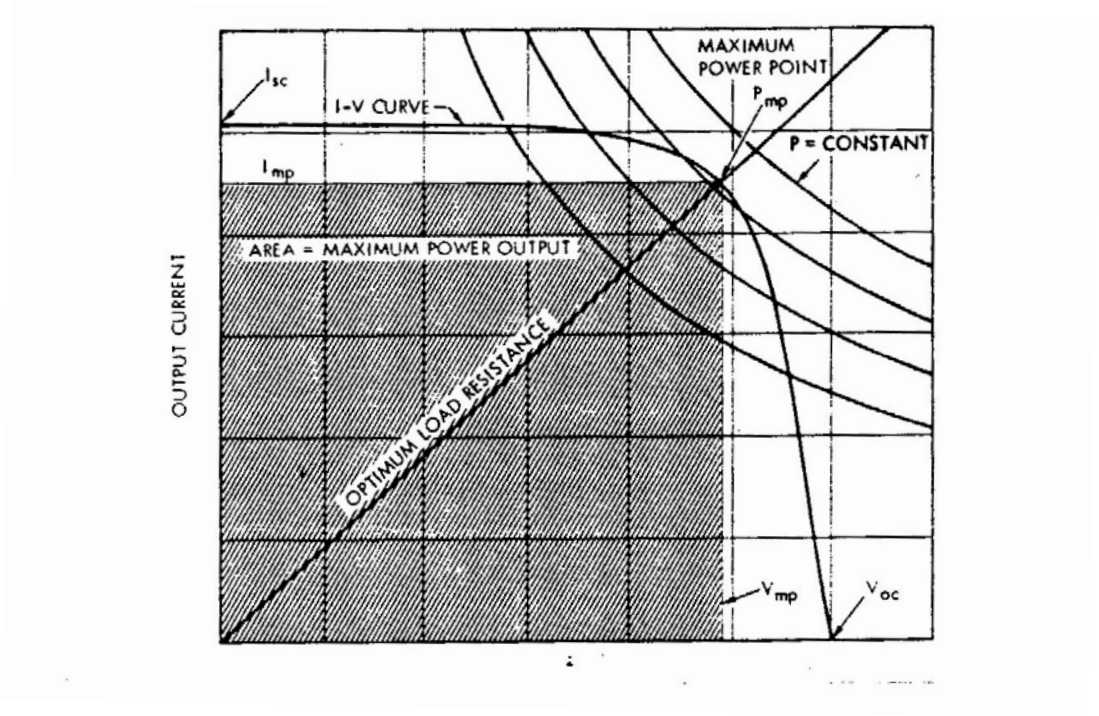

Figure 15. *I-V* curve showing P_{max} with V_{mp} and I_{mp} [from 13]

Efficiency of a solar cell is given by

$$\eta = \frac{P_{\max}}{P_{in}} \times 100 \quad \text{for efficiency in percent}$$

where P_{in} is the radiated power striking the cell area from the light source [13]. For this thesis, the solar spectrum is used as the light source. Air Mass Zero (see Figure 16 [15]) is the solar spectrum that reaches Earth's orbit and is used as the space light intensity and spectrum standard. To be consistent with industry, $135.3\,\mathrm{mW/cm^2}$ was used for P_{in} [16]

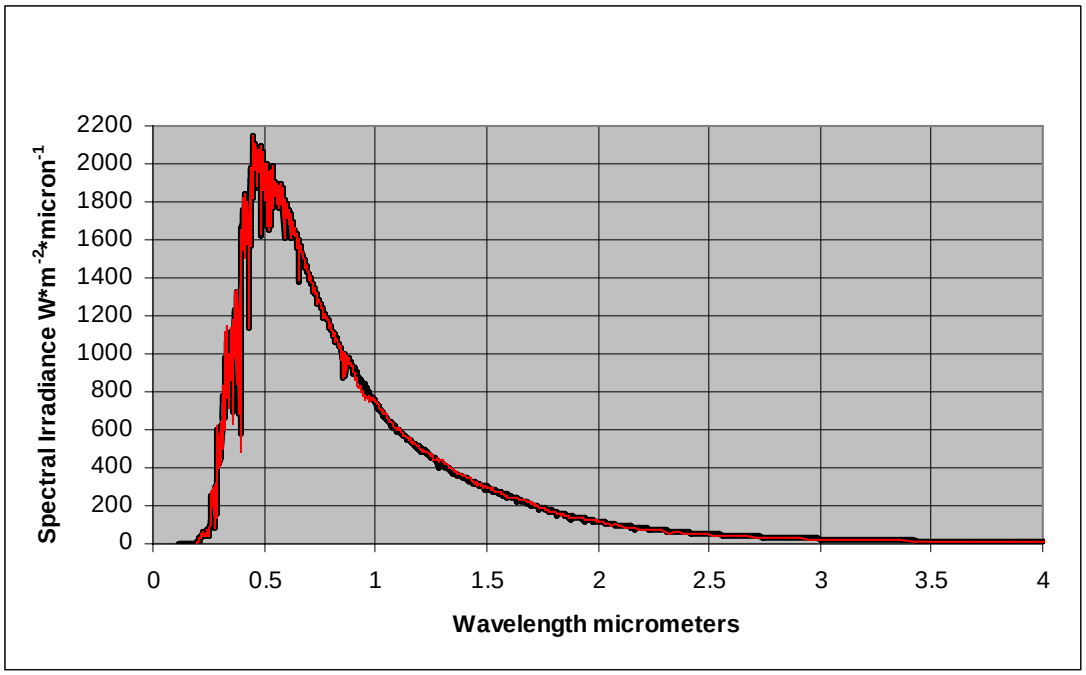

Figure 16. ASTM E-490 and Wehrli 1985 Air Mass Zero spectrum [after 15].

Another parameter sometimes used is Fill Factor (FF). Fill Factor is defined as:

$$FF = \frac{P_{\max}}{I_{SC} V_{OC}}$$

FF was not used as a comparison parameter in this thesis.

The most crucial factors of a solar cells performance is the band gap energy and the input light intensity. The band gap energy determines the minimum energy required to free a valence electron from its atomic bond. Therefore, the band gap energy is the amount of energy that a photon must impart to an electron to generate an electron-hole pair. The energy needed:

$$Energy = \frac{hc}{\lambda}$$

where *h* is Planck's constant, *c* is the speed of light, and λ is the wavelength of the light. Thus, a solar cell will only produce power for certain wavelengths of

light [3]. This limited response only to specific light spectrum led to the development of multi-junction solar cells to broaden the spectrum that a cell could fully employ.

C. MULTI-JUNCTION SOLAR CELLS

Multi-junction cells use different means to better utilize the AM0 spectrum. A multi-junction cell stacks layers of materials that have varying band gap energies to provide for more photogeneration. The physical stacking of one cell on top of the other created a parasitic junction between the two cells. This problem was recognized and a highly doped region was incorporated to create a tunneling junction to solve the problem (see Figure 17).

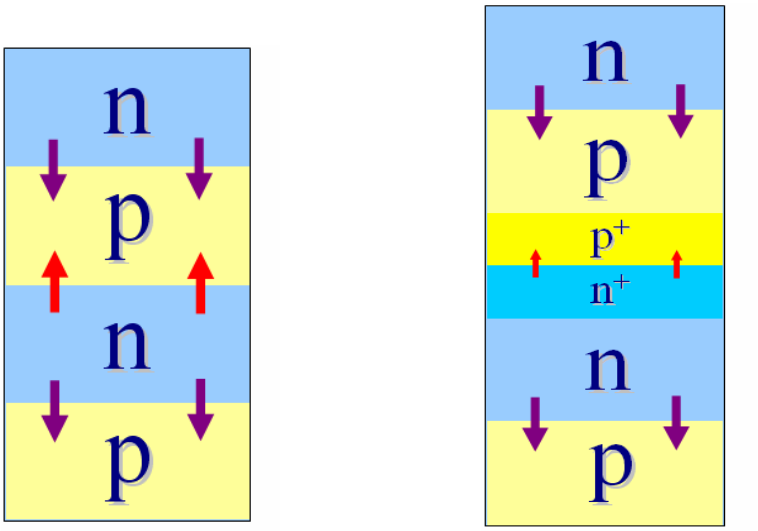

Figure 17. An example of physically stacking of solar cells showing (a) parasitic junction and (b) tunnel junction to minimize parasitic issue [after 3].

Ideally, the same material that had a means to vary its band gap energy would be used. Garcia demonstrated in Silvaco that Indium Gallium Nitride could be used as such a material [17]. By varying the mole fraction of InGaN, he was able to generate materials with various band gap energies. This varying material could be layered into a multi-junction cell. The difficulty with this work has been formulating the physical material into an actual solar cell. The material scientists

have had difficulty making a p-type InGaN material. Once these physical problems are overcome, a true multi-junction solar cell made from one material could be feasible.

Since the single material solar cell has not been feasible, engineers looked to stacking different materials to make better use of the input spectrum. The InGaP/GaAs/Ge triple junction cell appeared from this research. The typical band gap energies of InGaP is 1.902eV, of GaAs is 1.424 eV, and of Ge is 0.664eV [26]. These band gaps cover a wide range and permit better use of the incoming light spectrum (see Figure 2). A material between the GaAs and Ge, such as the 1.12eV Si, would produce a more efficient solar cell. In this case, the mismatches of the Si and GaAs lattice structures have prevented this improved cell. Again, the material scientists are attempting to solve a solar cell problem. InGaNAs has shown promise to solve this mismatch and cover this neglected part of the spectrum (see Figure 18). A quad junction cell could then be realized.

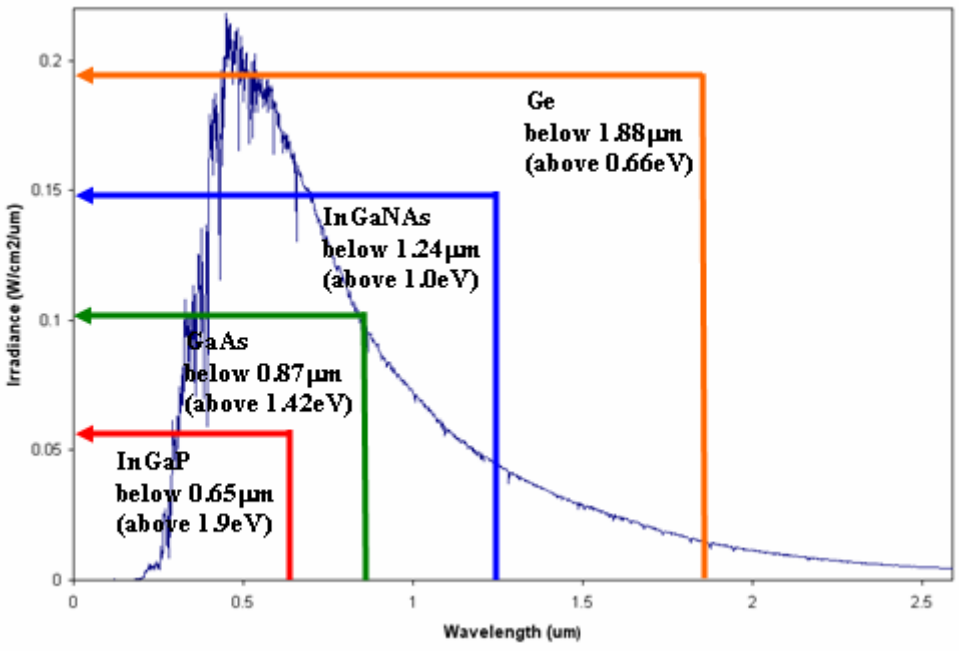

Figure 18. AM0 spectrum with photogeneration ranges for InGaP, GaAs, InGaNAs, and Ge [from 3]

III. TEMPERATURE EFFECTS ON SOLAR CELLS

An increase in a solar cell's operating temperature causes a slight increase in the I_{sc} and a significant decrease in V_{oc}. The increase in I_{sc} is typically small, usually in the range of tens of micro amps/^{o}C-cm^2. The change in voltage due to temperature is more significant at about 2mV/^{o}C. These changes will be addressed as each has an effect on the power and efficiency of the solar cell. The changes in V_{oc} contribute the majority of the changes is efficiency [18].

A. VOLTAGE

The main temperature dependence of a solar cell arises from the variations of V_{oc} with temperature. The changes in voltage result from the balance between direct, indirect, and Auger recombination rates of the carriers and the photogeneration of electron-hole pairs [19]. The change in the band gap also has an effect on the voltage. The first factor to be addressed is the temperature dependence of V_{oc} based band gap changes.

For most materials, the band gap decreases as temperature increases. The narrowing of the band gap creates a reduction in the V_{oc}. The result is a reduction in the cell's efficiency. The following equation relates the semiconductor's band gap energy as a function of temperature:

$$E_g(T) = E_g(0) - \frac{\alpha T^2}{T + \beta} \quad [20]$$

where $E_g(0)$ is the band gap energy at zero temperature, T is the temperature, and alpha and beta are the coefficients for band gap temperature dependence for a material. The change in band gap due to temperature has a large effect on cell performance, but electron-hole recombination contributes to the overall voltage decrease.

Green uses the three types of recombination in an in-depth physics-based model to show the relationship between changing V_{oc} and temperature [19]. A simplified result of his derivation showing the relationship of V_{oc} to temperature was:

$$\frac{dV_{OC}}{dT} = -\frac{\left(\frac{\left\langle E_{go} \right\rangle}{q} - V_{OC} + \frac{kT}{q} \left\langle \gamma \frac{f d\xi}{\xi df} \right\rangle \right)}{T} \quad [19]$$

Where V_{oc} is the open circuit voltage, $\xi = np \exp\left(\frac{-E_g}{kT} \right) / n_{ie}^2$, E_g is the band gap appropriate of the recombination process of interest, f is a general function in the limiting cases used in Green's paper, $E_{go} = E_g - T\left(\frac{dE_g}{dT} \right)$, T is the temperature, and $\frac{kT}{q}$ is the thermal voltage. Reference Green's paper for more detail. The significance of this equation was the approximately linear temperature dependence of V_{oc} with temperature. After addressing the second and third order effects, Green concluded that this equation is expected to by accurate for all solar cells, regardless of the recombination [19]. Green establishes the importance of the electron-hole product in recombination throughout the device, leading to a general formulation that temperature sensitivity is due to the open circuit voltage, accounting for 80-90% of the temperature sensitivity in the device [19].

B. CURRENT

Compared to the voltage, the short circuit current is not strongly temperature dependent. It tends to increase slightly with increasing temperature because the semiconductor band gap decreases with temperature" [21]

In a highly doped semiconductor, band gap separation occurs where the conduction band is lowered by the same amount as the valence band is raised. In other words, the band gap energy decreases as temperature increases. This band gap narrowing is simulated in ATLAS by the effective intrinsic concentration:

$$n_{ie}^2 = n_i^2 \exp\left(\frac{\Delta E_g}{kT}\right) \quad [22]$$

where n_{ie} is the effective intrinsic carrier concentration, n_i is the intrinsic carrier concentration, ΔE_g is the variation in band gap energy, T is temperature, and k is Boltzmann's constant. In the next chapter, the Silvaco simulation model, BGN, takes into account the band gap narrowing as temperature changes to provide a more accurate result. These effects can be described by relating the band gap variation, ΔE_g , to the doping concentration, *N*, by the expression:

$$\Delta E_g = BGN \bullet E\left(\ln\frac{N}{BGN \bullet N} + \left[\left(\ln\frac{N}{BGN \bullet N}\right)^2 + BGN \bullet C\right]^{\frac{1}{2}}\right)$$

As the band gap decreases, the solar cell responds to longer portions of the spectrum. Thereby, more electrons are able to receive the necessary band gap energy to generate an electron-hole pair. More photogeneration means more current is produced. Therefore, Isc increases as the temperature increases [18]. A general trend is decreasing band gap with increasing temperatures leads to more I_{sc} [19].

C. OTHER EFFECTS

Another small effect is as the temperature increases, the cell resistance increases. The resistance change is due to the mobility changes in the material as temperature changes. Figure 19 shows the general changes in the mobility as temperature changes.

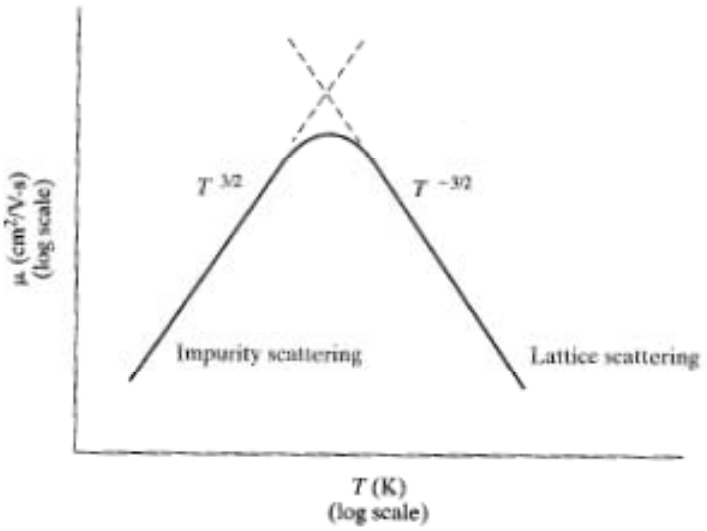

Figure 19. Approximate temperature dependence of mobility [from 9].

This mobility shift would change the conductivity, and thus, the resistivity of the device by the following equation:

$$\rho = \sigma^{-1} = q\left(\mu_n + \mu_p\right) n_i \quad [9]$$

where ρ is the resistivity, σ is the conductivity, q is electron charge, n_i is the intrinsic carriers concentration, and μ is the mobilities for electrons and holes. A change in the device's resistance can be another source of error. Mobility was not altered for this thesis.

Temperature also effects donor (N_D^+) or acceptor (N_A^+) atom ionization as given by the following equation:

$$N_D^+ = N_D\left[1 - \frac{1}{1 + \frac{1}{g}\exp\left(\frac{E_D - E_F}{kT}\right)}\right] \quad [20]$$

where N_D is the number of donor atoms, g is the ground-state degeneracy of the donor imputer level, E_D is the donor ionization energy, E_F is the Fermi energy, T is the temperature, and k is Boltzmann's constant. An analogous equation for acceptor atom ionization is state in Sze's book [20]. Figure 10 uses band

diagrams to illustrate the relationship between carrier concentrations and energies. The default Silvaco settings were utilized in this thesis. The models calculate the intrinsic carrier concentrations for each temperature. For most devices, it is preferable to control the carrier concentrations with doping vice thermally generated electron-hole pairs [9].

The changes in a material's index of refraction and extinction coefficient due to changes in temperature are addressed in the Silvaco modeling section of this thesis.

D. SUMMARY

For related, yet diverse, reasons temperature has a significant influence on the efficiency of a solar cell. The current output increases slightly but is relatively stable at higher temperatures, while the voltage is reduced. This combination causes an overall drop in power as the cell temperature is increased [23]. The combination of the two effects result in a general trend to loss in efficiency and power as temperature increases, mainly due to the decrease in voltage. A final word of caution about modeling these effects comes from Green:

> The importance of the electron–hole product in determining overall temperature sensitivity provides some constraints on appropriate expressions for modeling the performance of generic devices. Apparently sensible choices in this area can lead to errors in the modeled temperature sensitivities and an unintentional systematic bias in the conclusions from modeling [19].

E. CURRENT RESEARCH

For probes heading to the sun or Mercury missions, research into higher than normal operating temperatures for solar panels continues. Some of NASA's goals are to improve efficiency at high temperatures and improve lifetime at high temperatures. Some of the missions planned are the Mercury orbiter, operating at $450\,^{\circ}\mathrm{C}$ to the proposed Solar Probe, operating at $2300\,^{\circ}\mathrm{C}$ at four solar radii

[18]. Research into wide band gap material is being investigated due to the less degradation from high temperature. Figure 20 shows the shift in theoretical efficiency as a function of band gap.

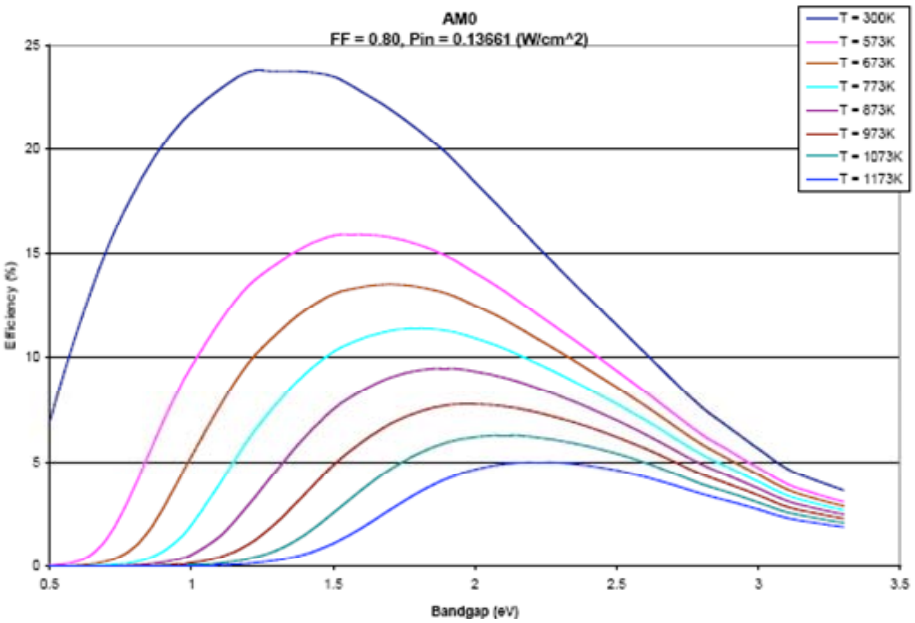

Figure 20. Theoretical efficiency of a solar cell as a function of band gap, showing the shift in optimum band gap [from 18]

New solar cells that can operate at high temperature are desirable; this requires development of high band gap semiconductors. A program to develop high temperature solar cells is in progress [18]

IV. SIMULATION SOFTWARE

Building upon previous work at NPS, the Silvaco software was selected to conduct this work. This section covers the software modeling and the strategy used to simulate a temperature dependent solar cell.

A. SILVACO INTERNATIONAL

Silvaco International is a company specializing in software modeling and simulation of semiconductor material. Their integrated TCAD suite of tools provides modeling and simulation capabilities for simple circuits to detailed integrated fabrication (see Figure 21 [24]).

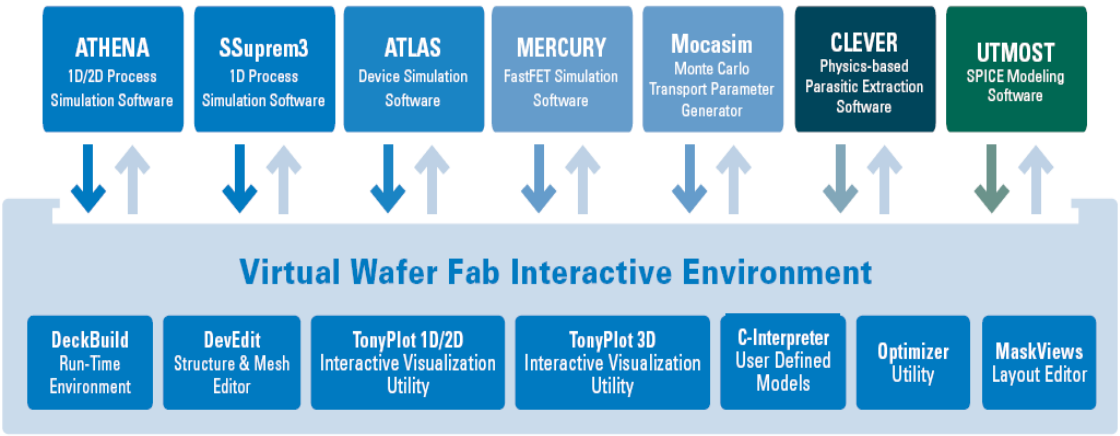

Figure 21. Silvaco Virtual Wafer Fab Integrated TCAD software [from 24]

1. Working with ATLAS

ATLAS is a core tool of the Silvaco VWF framework. It uses description files from either ATHENA or DevEdit or its own direct-input command files through DeckBuild. A combination of ATHENA and ATLAS makes it possible to determine the impact of processing on a device's electrical characteristics [22]. Figure 22 shows the information flow within ATLAS. The output, a Log file, was

used to evaluate the simulation and imported into MATLAB to generate various graphs. The scope of this thesis did not cover manufacturing processes nor address deficiencies based on fabrication, so only direct-input structure files into DeckBuild were explored. A declarative programming language states the desired structure for interpretation by ATLAS.

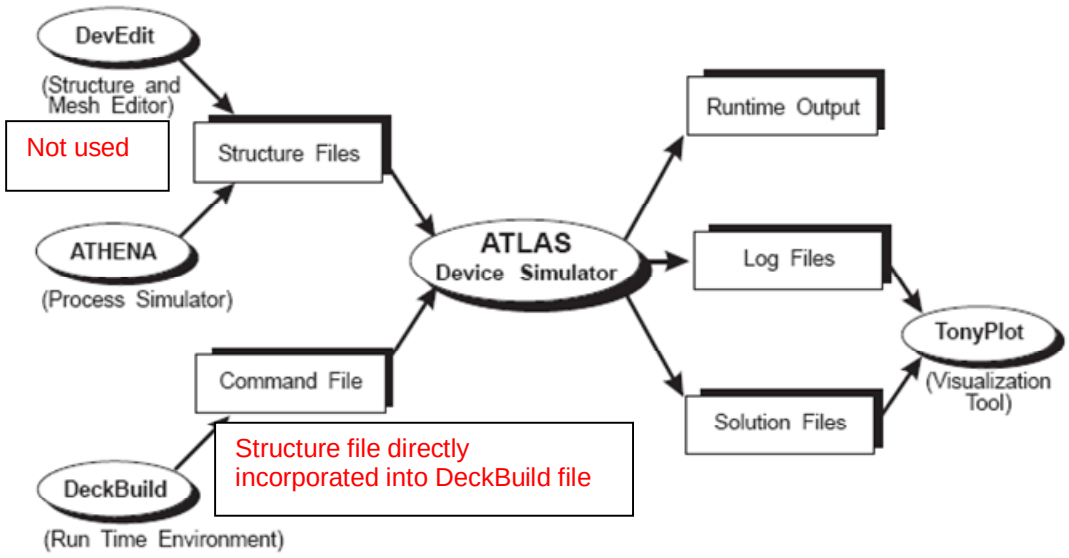

Figure 22. ATLAS Inputs and Outputs [after 22]

A brief structure description of the DeckBuild input files follows.

a. Constants

For the ease of changing various cell parameters, the input file lists cell constants as the first lines of code. This constant list was mainly an administrative exercise to ease the burden by permitting one simple change to the input deck without the requirement to find all occurrences of the parameter that was being altered. For further detail, when the constant is employed by program a "$" must proceed it to notify the program that it was previously defined. An example is shown below:

```
set cellWidth=5.000000e+002
set divs=1.000000e+001
set cellWidthDiv=$cellWidth/$divs
```

In this example, two constants are being defined, `cellWidth` and `divs`. These definitions are then used to calculate the new constant `cellWidthDiv` for later use in defining the spacing in the mesh.

b. Mesh

The mesh is specified onto which the device will be constructed. For this thesis, the mesh was defined using the ATLAS command language. The commands generate a gridline area used to define data points and solution points, similar to a finite element simulation. Two or three-dimensional figure can be constructed, comprised of many different sections (see Figure 23). To define a mesh using the command language, first the mesh dimensions must be stated followed by the spacing of the grid. Please note two important points about the mesh. One, is that the dimensions are in microns. Two, is that the vertical mesh is negative in order to build the solar cell above the surface of the device. This negative dimension is needed since the software interprets the positions, as a function of depth below the surface thus positive numbers would build the solar cell down into the device vice building it on top of the device. Rectangular or cylindrical coordinate systems can be used when defining the mesh. Constants can be defined as stated above for ease in defining and altering the mesh. ATLAS automatically adjusts the gridline spacing to match the desired value. The number of triangles in the mesh determines the resolution of the simulation. The density of the triangles is an important part of the simulation. If the density is too high then the execution time rises significantly without adding much to the resolution. If the density is too small, then the resolution is poor leading to an inaccurate or incorrect simulation.

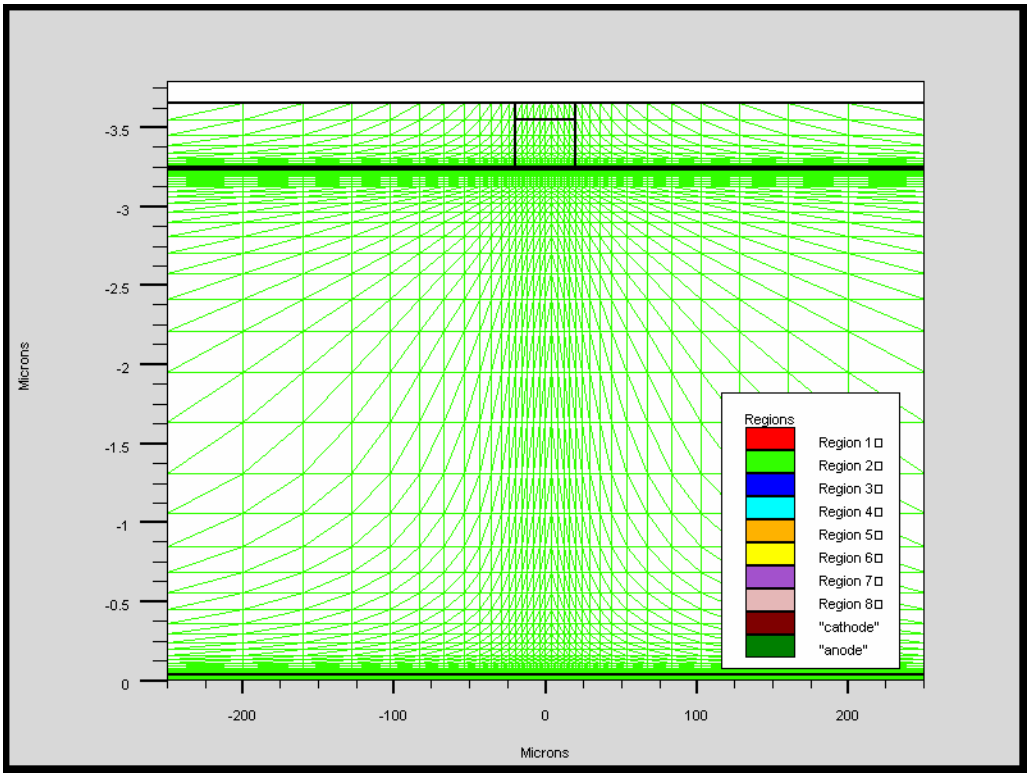

Figure 23. Generic Mesh for a GaAs cell

c. Regions

Next, material regions need to be specified. All parts of the mesh are assigned material names (see Figure 24). Specific materials are then later defined in the input file.

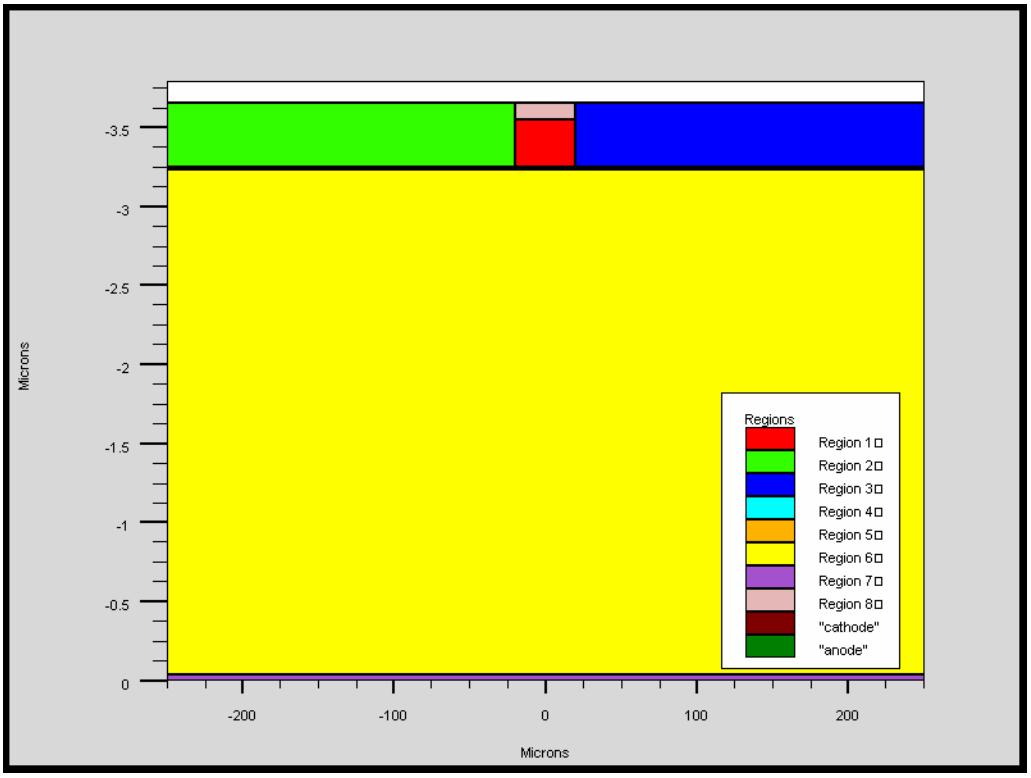

Figure 24. Regions for a GaAs solar cell

d. Electrodes

Electrical contacts must be specified in the ATLAS structure to obtain electrical properties (see Figure 25). Electrodes can be either defined as a specific material or ATLAS uses a perfect conductor at the specified location [22].

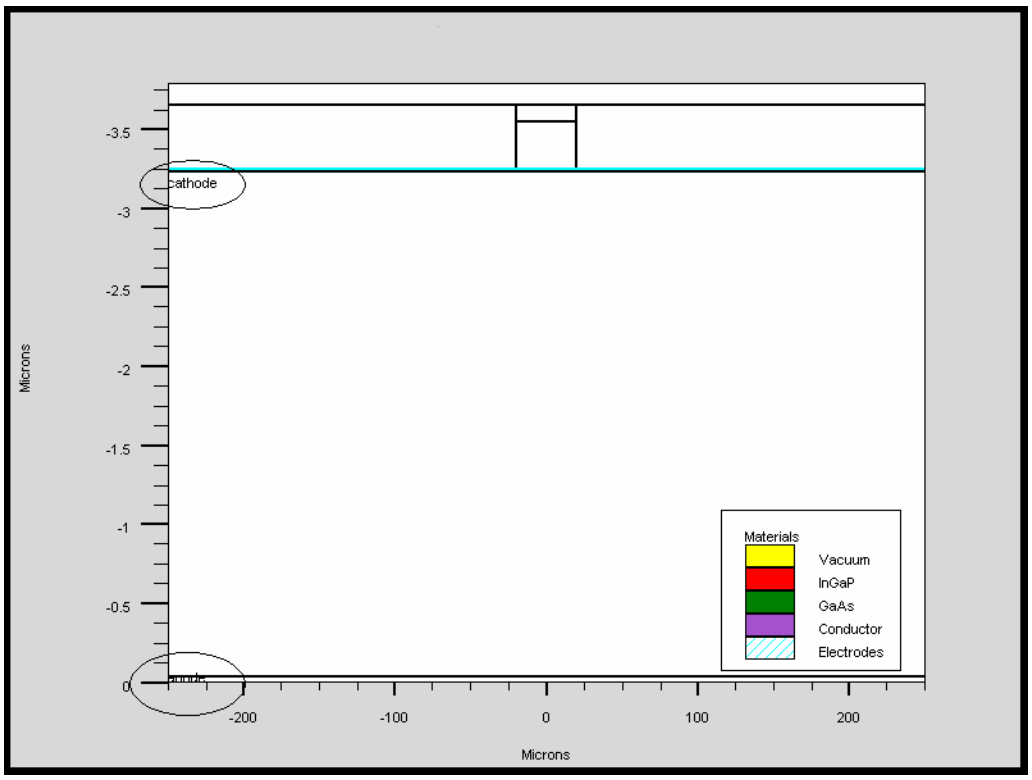

Figure 25. Electrode location

e. Doping

Each region with a semiconductor material is allocated a type and level of doping concentration. Doping can either be n or p type with a choice of uniform, linear, or Gaussian distribution. The concentration is in units of impurities per cubic centimeter. Figure 26 shows a typical uniform distribution of a GaAs single junction cell. Figure 27 is a close up depiction of the n-p junction to represent the differences in concentrations.

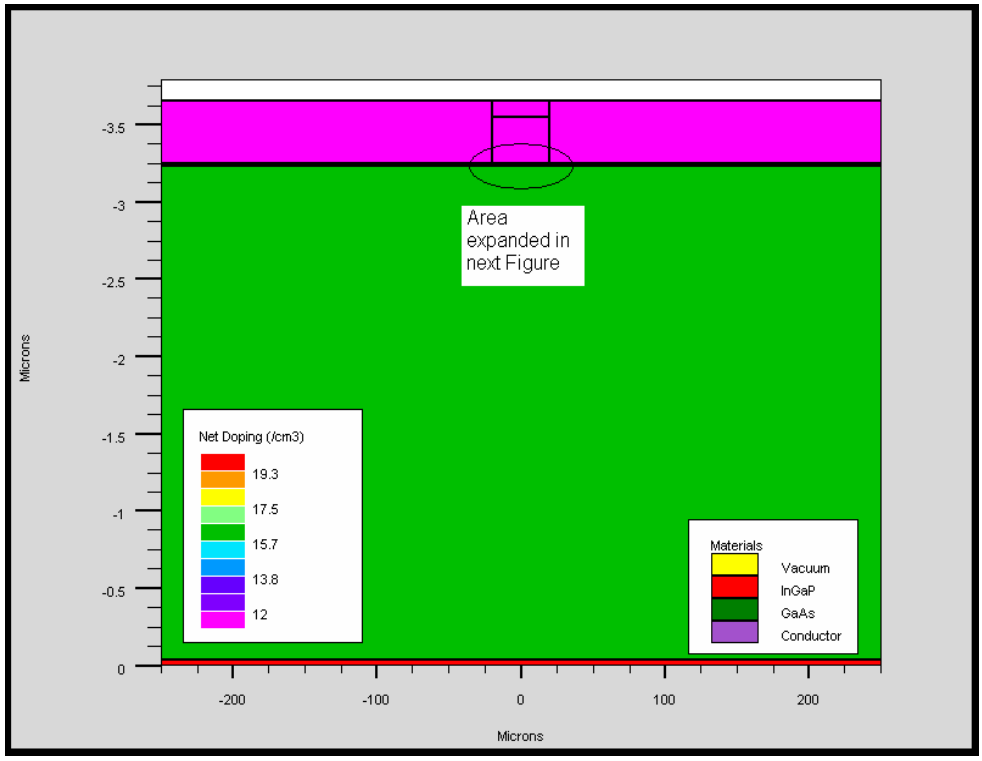

Figure 26. Typical uniform distribution for a GaAs single junction solar cell

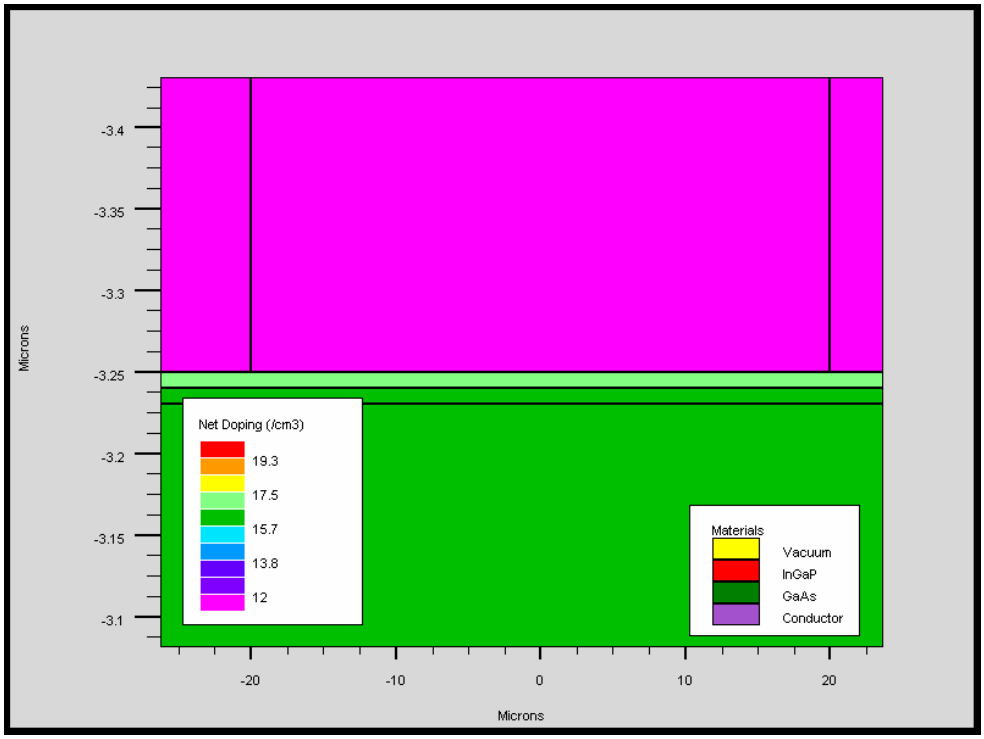

Figure 27. Close up of Figure 24 junction region to highlight doping changes in materials

f. Materials

A library of materials is a part of the ATLAS tool. Many common materials can be selected from this library for use in defining material properties. Since solar cells are using state-of-the-art materials that may not be listed in the library, ATLAS has the ability to fully define new materials.

A minimum set of property data must be specified for a new material to include band gap, dielectric constant, electron affinity, densities of conduction and valence states, mobilities, recombination coefficient, and an optical file containing refractive indices *n* and extinction coefficients *k* for a material [22]. The optical file determines the transmission and attenuation of light as it passes through the semiconductor. The optical files for the material used in this simulation were generated by an interpolation routine written by Michalopolous [5]. The temperature dependence of *n* and *k* will be discussed later.

g. Models

ATLAS can use many different models in a device simulation. These models can be defined for particular regions or over the entire device. The optical recombination (OPTR) model was used extensively throughout these simulations along with the concentration dependent mobility (CONMOB) model. The CONMOB is only valid for GaAs and adjust the electron and hole mobilities for GaAs according to dopant concentration [22]. The OPTR determines the possibility that a photon is generated when an electron and hole recombine [22]. Green has shown that the OTPR model increase the accuracy of the solar cell simulation [6]. After reviewing Canfield's thesis [7] and the ATLAS Users manual [22], the band gap narrowing (BGN) model was added to the model list. BGN, along with device temperature, takes into account temperature dependent band gap narrowing [7], [22]. This phenomenon was further expounded by an email from Dr. R. Jones of Silvaco International. He demonstrated through a run-time

output file from ATLAS that the band gap energy is recalculated as temperatures are varied in the input file.

An example for 300K:

REGIONAL MATERIAL PARAMETERS:

Region	1	2	3	4	5
Material :	GaAs	AlGaAs	AlGaAs	SiO2	
Conductor					
Type	semicond.	semicond.	semicond.	insulator	metal
Band Parameters					
Eg (eV) :	1.42	1.8	1.8		

Same simulation but at 350K:

Band Parameters				
Eg (eV) :	1.40	1.78	1.78 [25]	

h. Light

ATLAS' luminous optical-electric simulation module has the ability to determine photogeneration at each mesh point, thus a number of light sources can be simulated with various changes in their location, orientation, and intensity. Based on the ATLAS Users Manual, Bates explained it in very simple terms as follows.

> The refractive index *n* is used by Luminous to perform an optical ray trace in the device. Differences in n values across material boundaries determine the rate of light transmission and reflection. By following the path of light from the source to a mesh point, Luminous is able to determine the optical intensity at that point. The extinction coefficient *k* is used to determine the rate of absorption and photogeneration (electron-hole pair generation) for the calculated optical intensity at each mesh point. Together, these simulations provide for wavelength-dependent photogeneration throughout a multi-junction cell [3]

Through further research by Bates, another interesting issue arose. As with the mesh, the programmer can determine the range of wavelengths to be used in a multispectral simulation. The programmer defines the number of wavelengths within that range as well. Again, as with the mesh, defining an

insufficient number of wavelengths to calculate can lead to an erroneous simulation. Bates went further to explain that a step size of 0.001 μm produced a stable and reliable outcome. Another issue that seemed to be resolved by the two test runs was conducted to test this theory. Figure 28 shows the *I-V* curve of a GaAs solar cell using a larger step size based on an early beam model by Bates. The beam step size is larger than 0.001 μm ,at about 0.0047 μm . Figure 29 shows the *I-V* curve of the same cell but using Bates' smaller step-sized beam. Note the I_{sc} on each. A typical state-of-the-art GaAs cell has an I_{sc} of approximately 30.5 mA/cm^2.

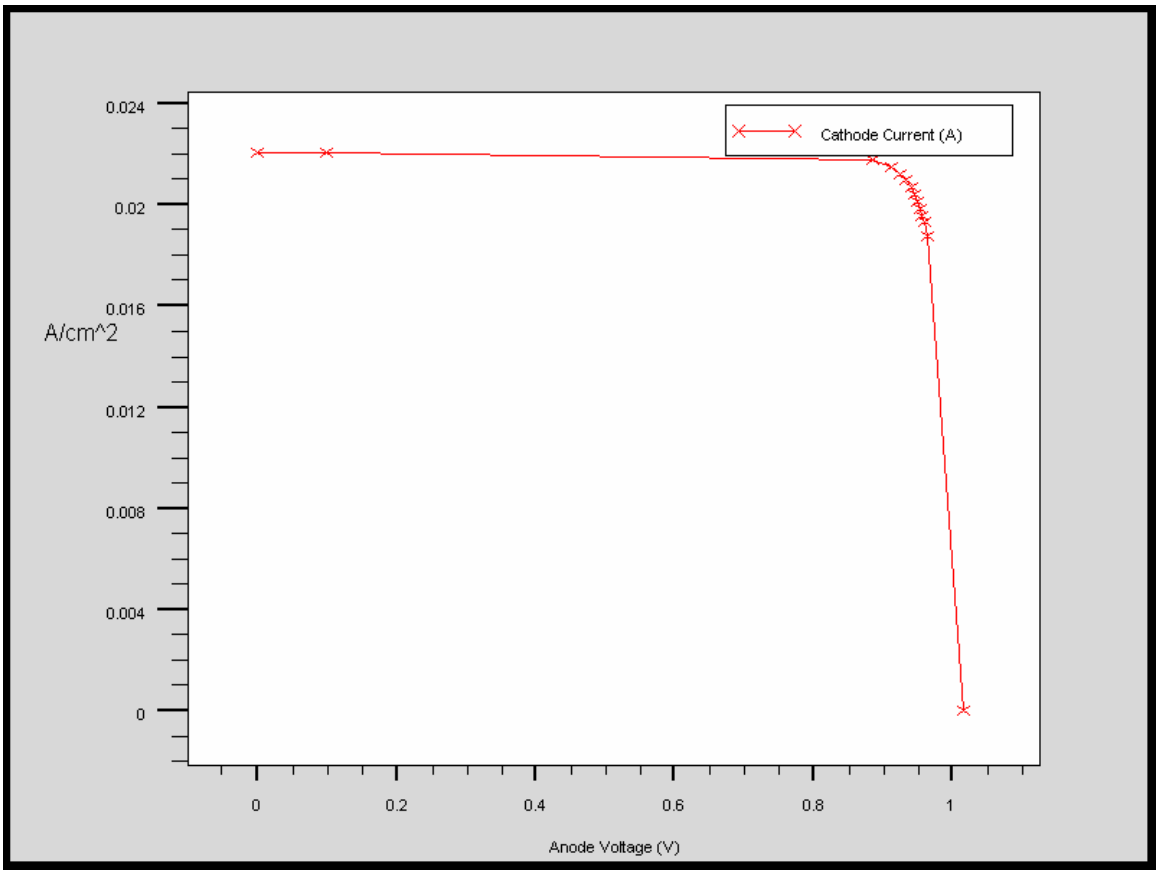

Figure 28. GaAs solar cell simulation using 0.0047 μm wavelength step size

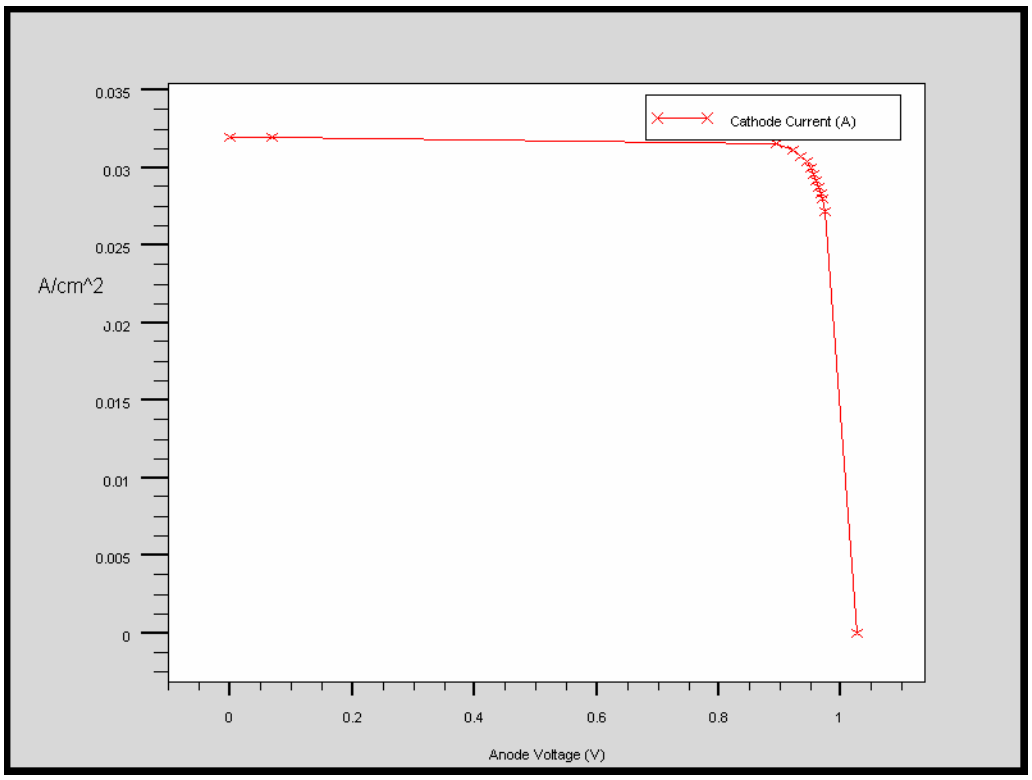

Figure 29. GaAs solar cell simulation using a 0.001 μm wavelength step size

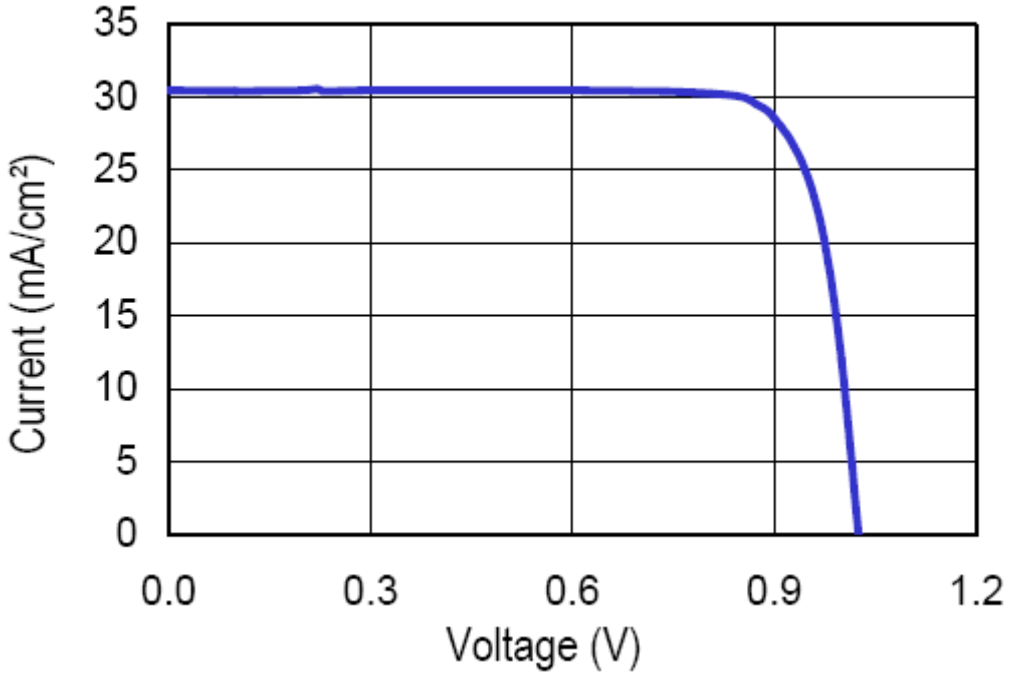

Figure 30. Typical *I-V* curve for a production GaAs solar cell [from 16]

While there is some error in both figures, smaller step-sized produced an *I-V* curve closer to a real solar cell design (see Figure 30). Green explains this discrepancy as:

> ...a statement in the ATLAS code of '`solve icathode=17.404e-8 b1=1.`' With the appropriate conversion of the current (multiplying by a scalar of 200,000) the current for this device solving statement is 34.808 mA/cm^2. It is normal to place such a statement before solving for the current being equal to zero to resolve convergence issues. This statement shows that the solution for the short circuit current was over the 27.6 mA/cm^2 that was reported and is actually very close to the simulated value from this thesis. Therefore, considering that the properties for GaAs are very well known in Silvaco, it is assumed that this analysis is correct [6].

The scaling factor that Green mentions does explain a possible source of error. It does not explain the discrepancy of Figure 28. Bates' step-size theory helps to explain the reduced current value in Figure 28. Further investigation into the simulation process would be needed to generate a definitive answer. For this thesis, Bates' smaller step-size was used to maintain consistency through test runs.

For later simulations, another source of error could be the temperature dependence of index of refraction, *n,* and extinction coefficient, *k*. Since *n* and *k* are mainly obtained through empirical measurements, this thesis did not alter the *n* and *k* data even as temperature varied. The data for the more exotic material, like InGaP, could not be found, so it was determined the best course of action would be to not alter *n* and *k* for the any of the materials. This way, the results would be consistent, though some error is expected. This assumption was also based on the relative small changes in n and k over the temperature ranges simulated. In Figure 31 [26], the *n* value for Ge only varies by less than 0.04.

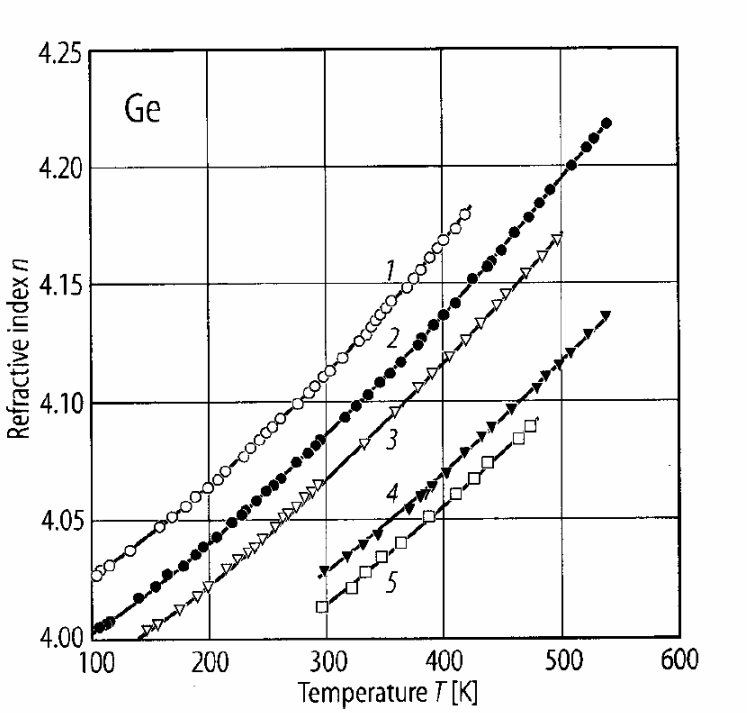

Figure 31. Ge temperature and wavelength dependence of the refractive index, *n*. Curve 1: λ=1.970μm; Curve 2: λ=2.190μm; Curve 3: λ=2.409μm; Curve 4: λ=3.826μm; Curve 5: λ=5.156μm [from 26]

The index of refraction for GaAs is given by the equation:

$$n = 3.255\left(1 + 4.5\times10^{-5}T\right)$$ [26]

For this value of *n*, table 1 calculated the changes in *n* for GaAs for the ranges of this thesis. The percent difference was about 0.33%.

T (K)	300	325	350	375
n =	3.2989	3.3026	3.3063	3.3099

Table 1. Index of refraction for GaAs over specified temperature ranges

Finally, Figure 32 [27] shows a graph of both *n* and *k* for GaAs. As can be seen, *n* varies by less than 0.04 and the change in *k* is negligible for the temperature range from 300K to 375K.

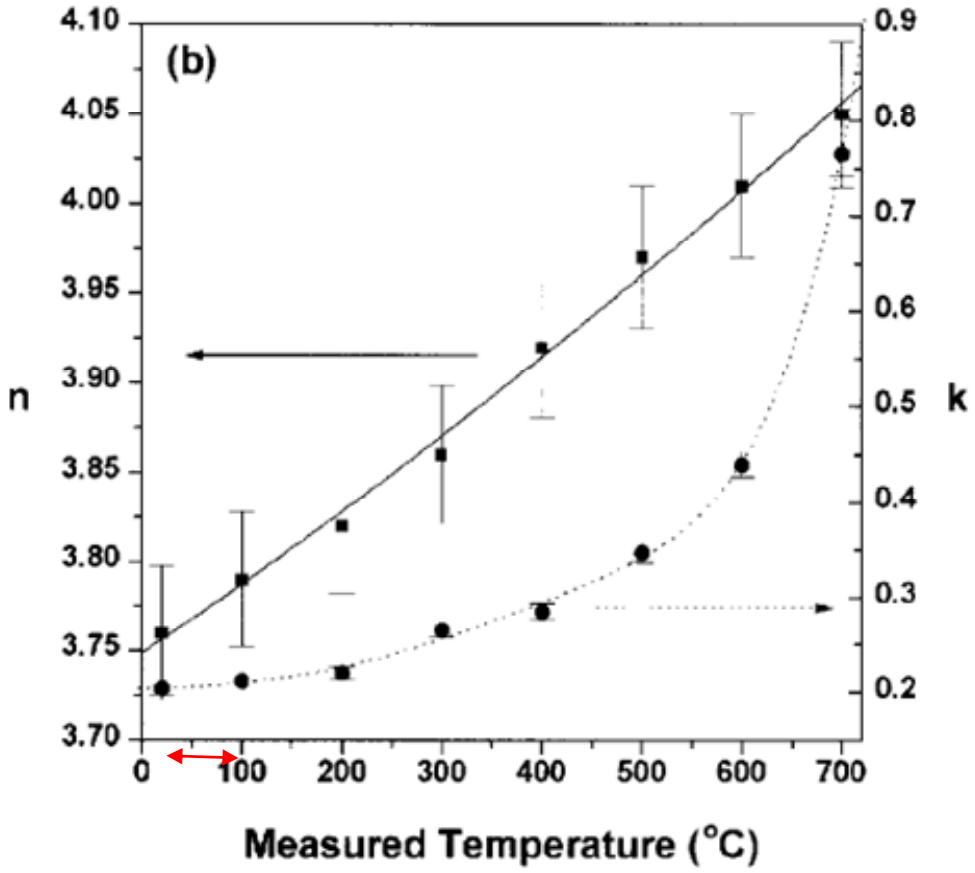

Figure 32. Real and imaginary parts of the refractive index (*n* and *k*) for GaAs for temperatures from 20-700°C [after 27]

No such examples were found for InGaP, but the above figures and data illustrate that the changes in *n* and *k* are small and can be assumed constant for this simulation.

i. Solving

After defining all input parameters, ATLAS can use various methods to solve for output data. Various numerical methods can be used to solve the cell parameters. The solution method was chosen after review of previous thesis, mainly Michalopoulos and Bates, based upon their author's results. The resulting solutions then can be used to generate I_{sc}, V_{oc}, and cell

efficiency, along with representative *I-V* curves and frequency responses. For this thesis, solutions of efficiency and plots of the *I-V* curves were used extensively.

The *I-V* curve provides a wealth of information about the solar cell. A series of solve statements in the ATLAS programming code generates an *I-V* curve. The solve statement `solve b1=0.9, b2=0.9` sets the first and second beams to 90% to take into account inefficiencies in the beams entering the cell. The number of times that the solve statement attempts to converge is set by the `itlimit=100` while the number of times that it attempts to solve the statement is set by `maxtraps=10.` The *I-V* curve is then generated through a series of statements such as:

```
solve ianode=-$i25 b1=0.9, b2=0.9
solve ianode=-$i24 b1=0.9, b2=0.9
solve ianode=-$i23 b1=0.9, b2=0.9
      ⋮
```

The result can be displayed using the Silvaco TonyPlot (see Figure 28 as an example) or imported into MATLAB for further investigation.

j. Simulation Code

A full set of source code is listed in Appendix A. In order to foster understanding, a simplified version is presented.

```
go atlas

# Definition of constants

# Mesh

# X-Mesh

# Y-Mesh

# Regions

# Electrodes
```

```
# Doping

# Material properties

# Models

# Light beams

# Solving
```

Each section is commented with code for that section. See Appendix A for code with comments.

B. INTERACTION WITH MATLAB

Even with the host of TCAD software, there was a need for more data flexibility then provided by TonyPlot or the output data "log" files. Several MATLAB codes from previous work were modified for use in an iterative process to generate results. From MATLAB, Excel spreadsheets were then employed to compare data in a simple format. From previous work with some modifications, the following files were used.

`filerw.m` [28]

Filerw.m was called to read a DeckBuild input file, modify stated parameters, and write the file for use by MATLAB GUI.

`mj_ivmaxp.m` [3]

Mj_ivmaxp.m file takes the output log file generated by DeckBuild and calculates the various output parameters and plots the *I-V* curve.

`maxpower.m` [3]

Maxpower.m solves for maximum overall power.

`eff_pmax_plot.m` and `eff_pmax_2Dplot.m` [28]

Eff_pmax_plot.m takes the log file and generates various plots to illustrate the results of the simulation.

`time.m` [28]

Time.m was used to tracking purposes to maintain the amount of time needed for each simulation.

`ATLASARUN.m` [28] with extensive modifications and insight for multiple loops taken from [7]

ATLASARUN.m was renamed `inter_test_XX.m` with various title changes to reflect the type of simulation being executed. The m-file was modified slightly in order to test thickness changes and doping changes, or to conduct single vice multi-junction cell tests.

While Canfield [7] was able to import his data directly into a spreadsheet using `xlscell.m,` this research was not able to get this automated tool to work. Therefore, all spreadsheets were taken from MATLAB sources and pasted into empty spreadsheets for comparisons. All MATLAB code is presented in Appendix B.

C. MODEL LIMITATIONS

Based on a solution of Maxwell's equations at each cross point in the mesh, ATLAS is a powerful simulator for electrical modeling, the accuracy of the solutions are dependent on the accuracy of the inputs, the mathematical methods selected and the reality of the simulation to the actual cell. Each of these compromises introduces error into the result. This error will be addressed in the following section

With minor changes, the multi-junction cell used by Bates was used as a baseline for comparison. The material properties and optical constants were assumed correct through Bates' research. The temperature variation of optics constants was not altered and was addressed in the temperature section of this thesis. Doping concentrations were uniform across an entire region with abrupt changes at boundaries. All materials were free of defects. Even though ATLAS has a capability to model an anti-reflective coating at the cell surface, a beam

intensity of 90% was used to account for light losses at the surface. Even with work from [29], a working tunnel junction model has not fully materialized using the Silvaco toolset. Aerospace Corporation has stated that they have developed a working tunnel junction model but sited proprietary reasons for not being able to share their model. Following [3] work, all tunnel junction regions were modeled as a vacuum with an adjusted refractive index to match the expected reflection at the junction. The results from this modified model required that all voltages and currents form individual cells to be externally combined using MATLAB to achieve a total solar cell solution.

D. SUMMARY

This overview of the simulation software hopefully demonstrated the capability of ATLAS and some of the inner workings of this testing. Even with the software limitations, ATLAS can properly simulation a multi-junction solar cell within a reasonable error. The work of Michalopous and Bates has shown sufficient accuracy in solar cell simulation. The next chapter shows the results obtained by altering the operating temperature of the cell models used in these previous works.

V. RESULTS

A. SINGLE JUNCTION CELLS

1. GaAs

A single junction cell based on [3] was selected to test whether Silvaco ATLAS could simulate a solar cell at various temperatures. All three materials, InGaP, GaAs, and Ge, were tested but only GaAs is presented for review with similar results for each part of the triple junction cell. All source code is presented in the Appendices.

As to be expected, the voltage decreased with increasing temperature. Voc nominally drops about two mV/$^{\circ}$C [13]. Figure 33 depicts the effects of temperature on a single junction GaAs cell.

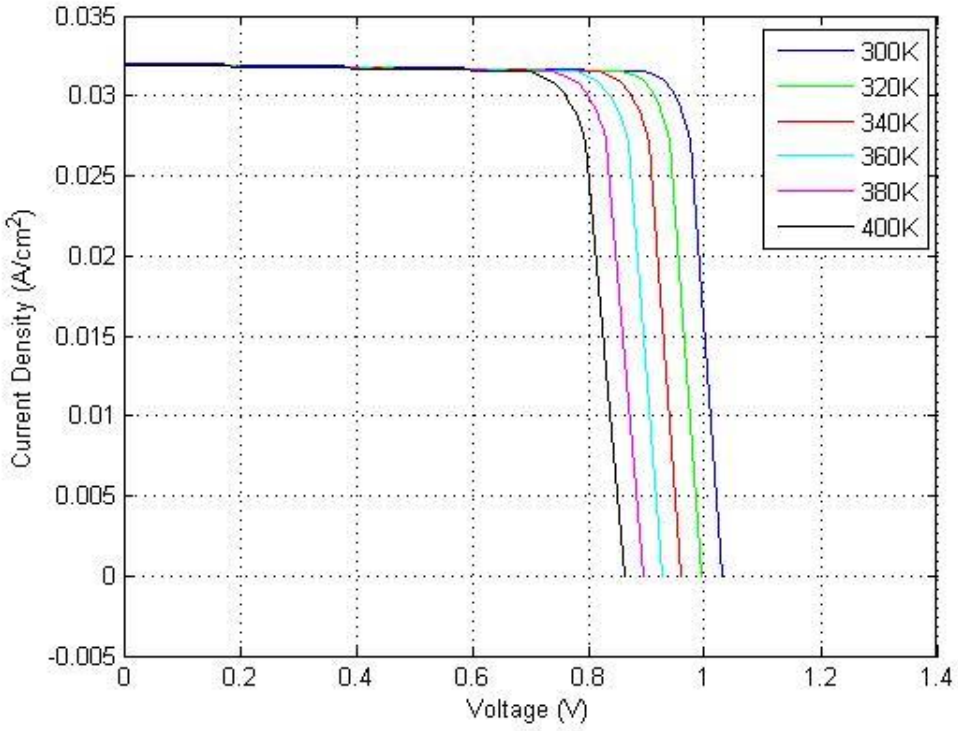

Figure 33. Simulated *I-V* curve for GaAs single junction cell with temperature variation

Figure 34 plots the efficiency versus temperature for all three single junction cells, illustrating a drop of a little less than 1% in efficiency per 20-degree change for GaAs. This linear relationship is typical of Green's physical model presented in section III-A, that voltage and temperature vary linearly. With many satellites operating in the 340K and higher, a substantial loss in efficiency was observed.

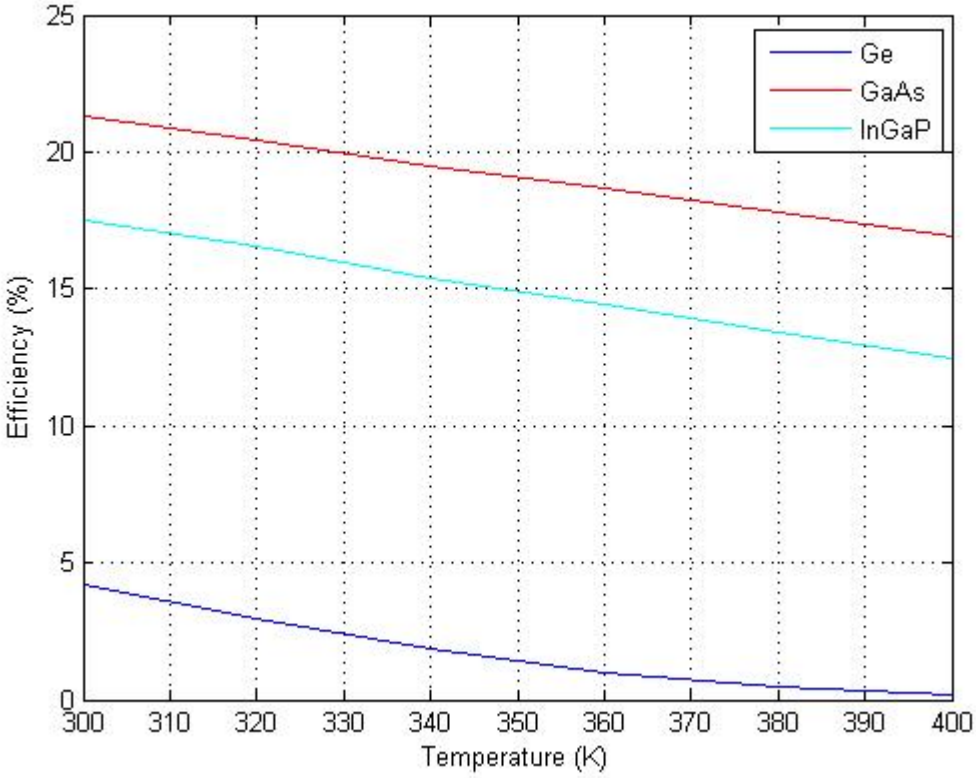

Figure 34. Efficiency in a single junction cells due to temperature variation

The other two materials, InGaP and Ge, had similar drops in voltage and efficiencies. To note an interesting observation, the Ge single junction cell had substantial losses at higher temperatures due to its low output voltage.

2. Ge

Due to the time limitations of this paper, the following observation is made but further investigation will not be presented. The Ge single junction cell had substantial losses at higher temperatures, bringing into question the need for such an inefficient cell (see Figures 35 and 36).

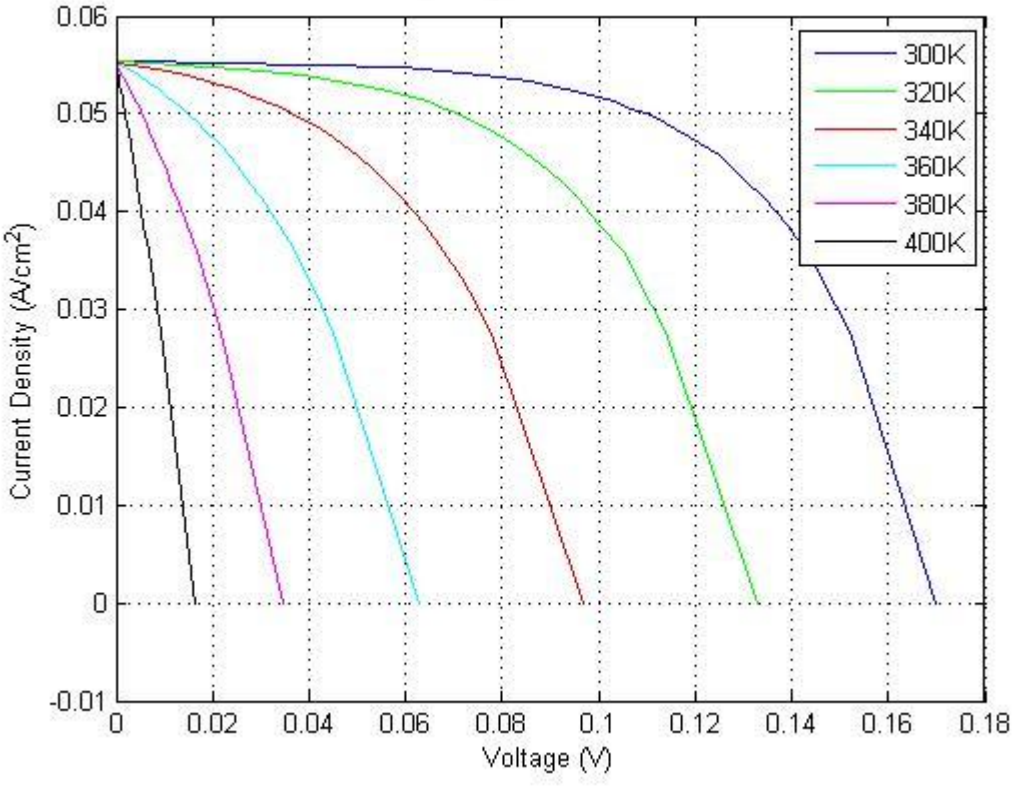

Figure 35. *I-V* curve for Ge single junction cell with temperature variation

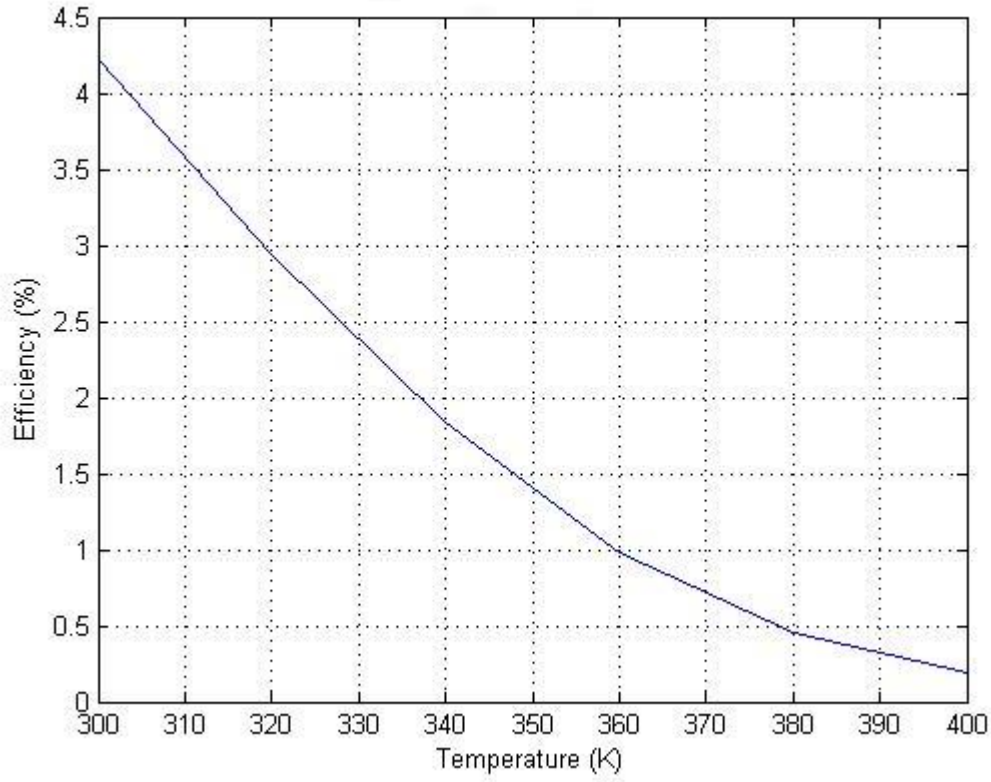

Figure 36. Efficiency in a single junction Ge cell due to temperature variation

Satellites are notoriously power hungry, so even at high operating temperatures, the small contribution of the Ge cell may have benefit. The benefit would have to be compared to the increase in manufacturing costs to include such an inefficient Ge cell.

B. TRIPLE JUNCTION INGAP/GAAS/GE CELL

Figure 37 shows the baseline cell from [3] that was simulated as a starting point.

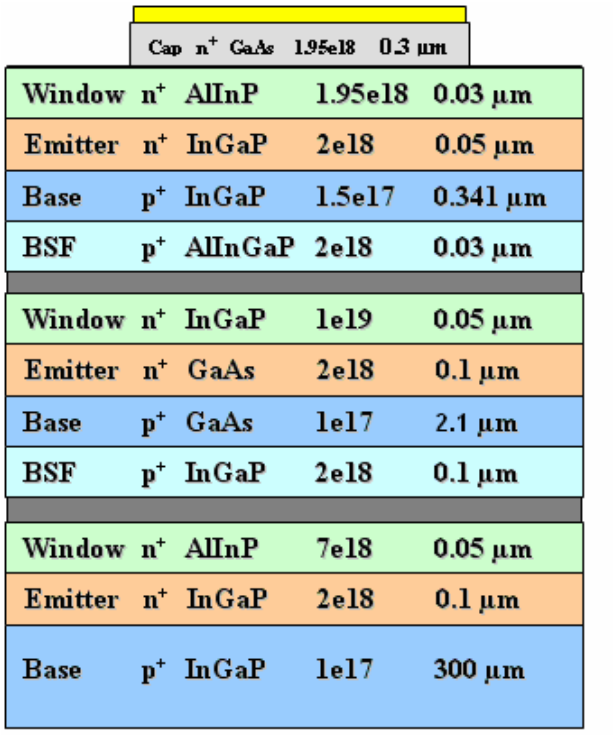

Figure 37. Baseline triple junction cell [after 3]

The above cell was simulated using ATLAS resulting in the following *I-V* curve (see Figure 38). The more closely matched Ge current resulted in a slight improvement over Bates' cell.

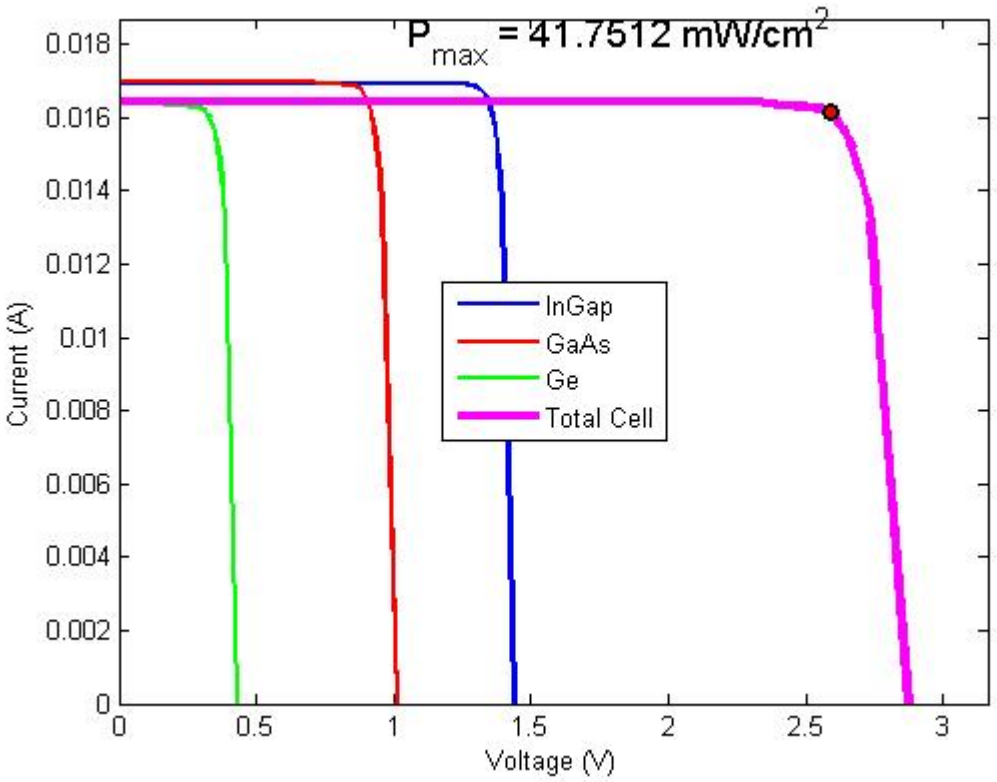

Figure 38. *I-V* curve for triple junction cell at 300K with maximum power displayed

Next, the same cell was simulated at different temperatures by changing the temperature by 25K. Figures 39, 40, and 41 show the *I-V* curves of the solar cell model output at the temperature intervals.

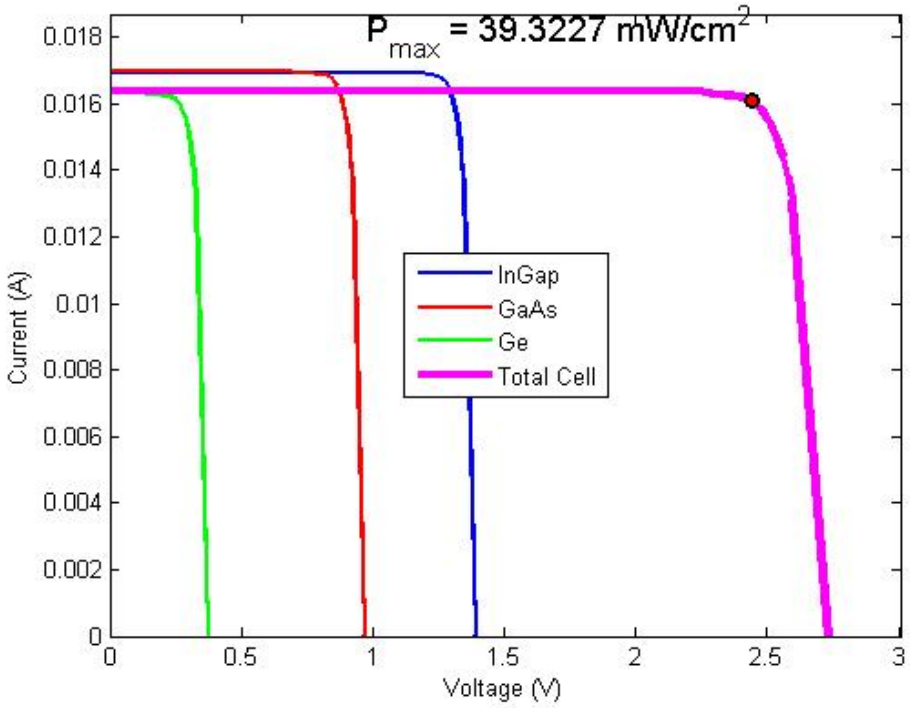

Figure 39. *I-V* curve for triple junction cell at 325K with maximum power displayed

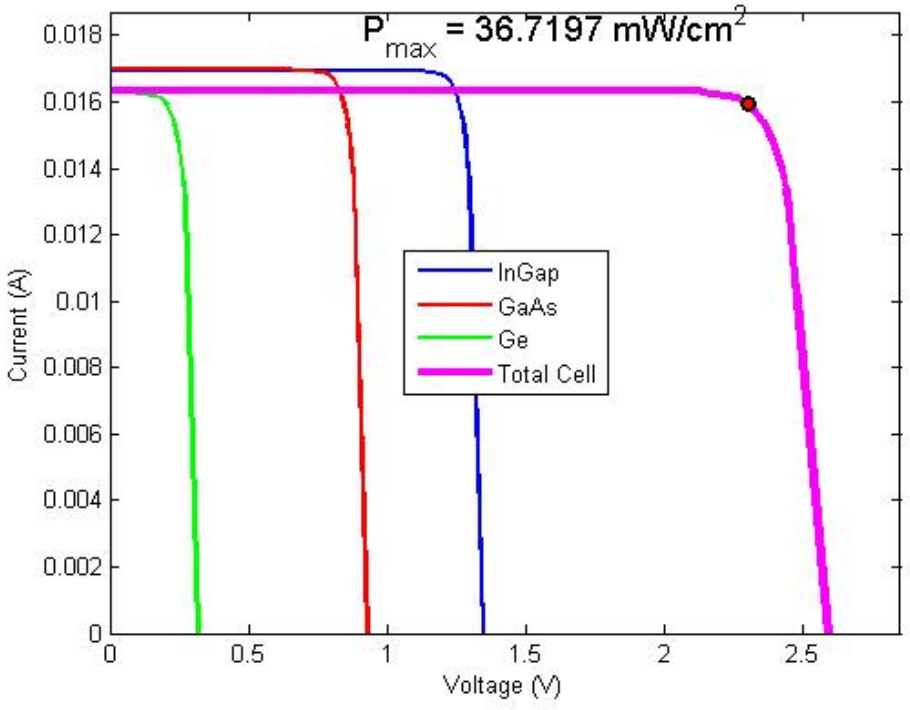

Figure 40. *I-V* curve for triple junction cell at 350K with maximum power displayed

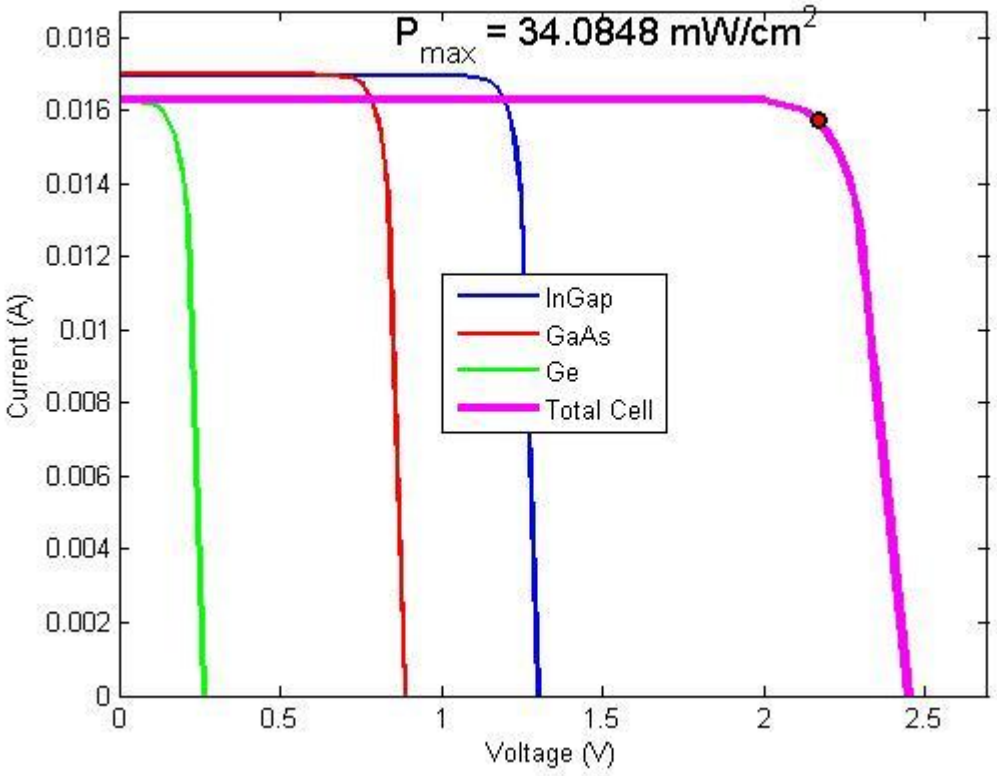

Figure 41. *I-V* curve for triple junction cell at 375K with maximum power displayed

As the above *I-V* curves show, the overall power drops as the temperature increases. From Chiang's experimental result that shows a drop of 5.8 mV/°C (see Figure 42), the temperature effects on the model output were compared favorably with the actual experimental data of a triple junction cell [30]. Table 2 and Figure 43 were developed to compare the experimental and simulated results. The percent difference in the data was less than 0.2%. Given that the [30] solar cell was only rated at 23% efficient, a state-of-the-art triple junction cell was given the same comparison test. The second test was based on [16] that has a beginning of life temperature coefficient of -5.9 mV/°C. Again, the results are promising at about 0.28% error (see Table 3).

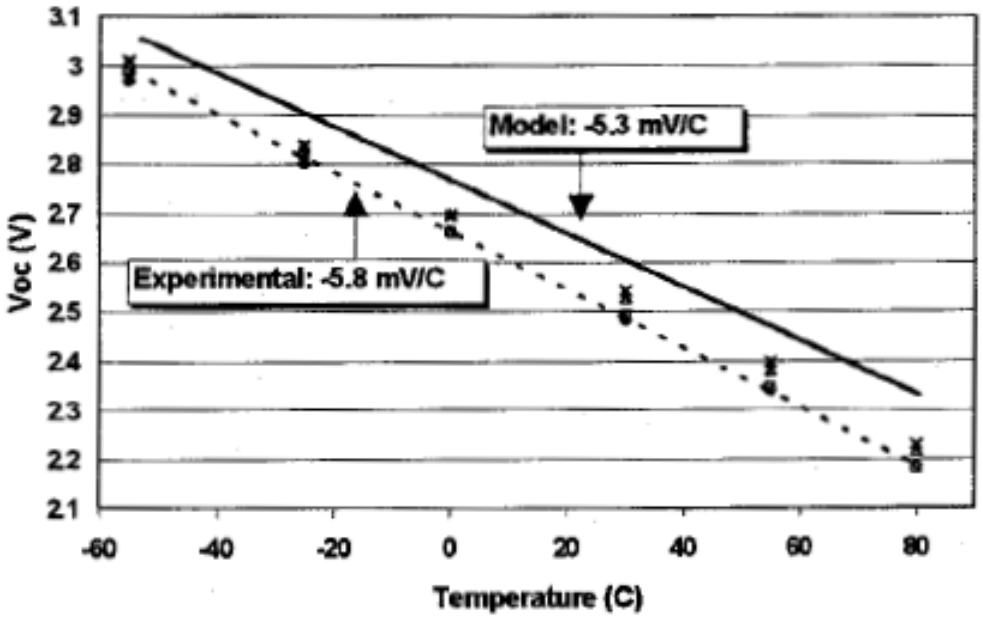

Figure 42. V_{oc} for a triple junction cell as a function of temperature [from 30]

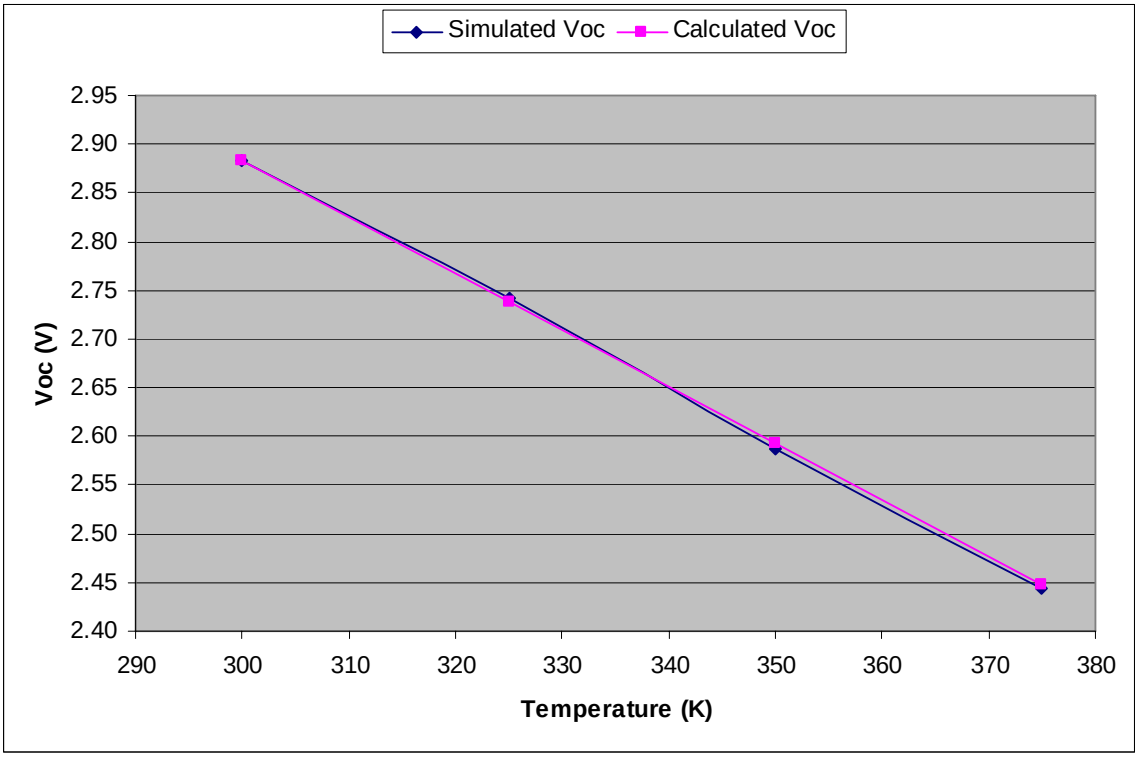

Figure 43. Comparison between calculated and simulated V_{oc} using [30] temperature coefficients

Original Simulated Voc at 300K	Temperate (K)	Calculated Temperature Dependent Voc (using -5.8mV/°C)	Simulated Temperature Dependent Voc (V)	Difference between Calculated and Simulated (V)	Percent difference between Calculated and Simulated Voltages
2.882	300				
	325	2.737	2.743	0.005	0.189%
	350	2.592	2.587	0.005	0.195%
	375	2.447	2.445	0.003	0.109%

Table 2. Comparison between calculated and simulated V_{oc} using [30] temperature coefficient

Original Simulated Voc at 300K	Temperate (K)	Calculated Temperature Dependent Voc (using -5.9mV/°C)	Simulated Temperature Dependent Voc (V)	Difference between Calculated and Simulated (V)	Percent difference between Calculated and Simulated Voltages
2.882	300				
	325	2.735	2.743	0.008	0.281%
	350	2.587	2.587	0.000	0.002%
	375	2.440	2.445	0.005	0.198%

Table 3. Comparison between calculated and simulated V_{oc} using [16] temperature coefficient

With this promising verification, an optimization technique was needed to see if a more efficient solar cell could be built.

After review of [3], [5], and [7] and with only the programming ability to vary two parameters, the InGaP and GaAs junctions were selected for further optimization. Based on [7] the parameters were narrowed to only the emitter and bases of the two junctions. A major obstacle that [7] did not deal extensively with the interactive effects of the layering of multi-junction cells. To optimize one junction at a time would be misleading due to the interactive effects the various junctions have on the cell [3]. This thesis was not able to optimize one junction at a time due to the assertion that the top cells have an obvious effect on the performance of the bottom layers. To understand the scope of the problem involved, a short digression is made.

Each junction has four separate layers (see Figure 37) that have two independent variables that were analyzed, thickness and doping. With the assumption that the bottom Ge layer remains static, there are eight variables to be changed in order to optimize the cell. To further investigate a better optimized

solution, one parameter would be held steady while all other parameters were iterated over a range in order to narrow to an optimal solution. Then this process would have to be repeated for each variable. Selecting six iterations leads to 6^8 or 1,679,616 possible variations. Bates' research used a genetic algorithm to vary many parameters in order to optimize a multi-junction solar cell [3]. Time restricted this thesis to a more simplified optimization, similar to [5], [6], and [7], using only two variables over four temperature ranges.

With a baseline established, using the MATLAB code `inter_test_base_mj.m`, variations to base thicknesses were iterated to observe the effects of changing the base thicknesses. Next, a successive run was conducted by changing temperature in the DeckBuild input file to 325K. Finally, the same process was done at 350K and 375K. The original base values were 0.34 and 2.1 μm for the InGaP and the GaAs bases respectively. The base layers were varied from broadly 0.22 to 0.52 μm for the InGaP base to 0.5 to 4.5 μm for the GaAs base. The results of varying the temperature produced a cell that seemed to have the same dimensions as the baseline cell (see Figures 44 thru 47). Table 4 lists the best results from each test. As is highlighted the optimum cell had the same base thickness as the cell designed to operate at room temperature.

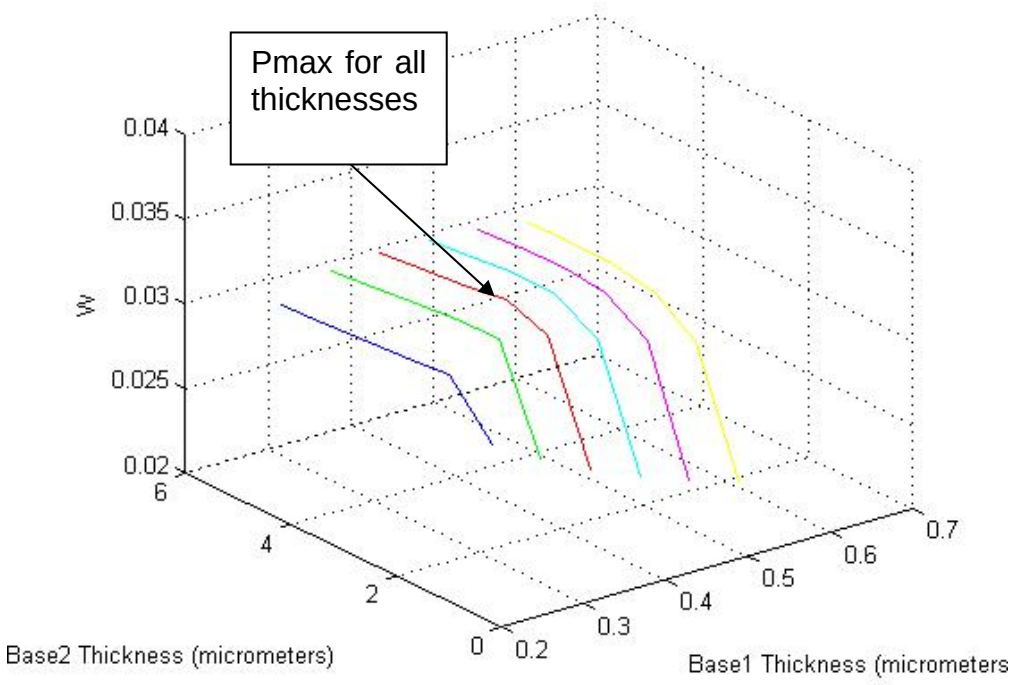

Figure 44. Iteration test for different bases thicknesses at 300K

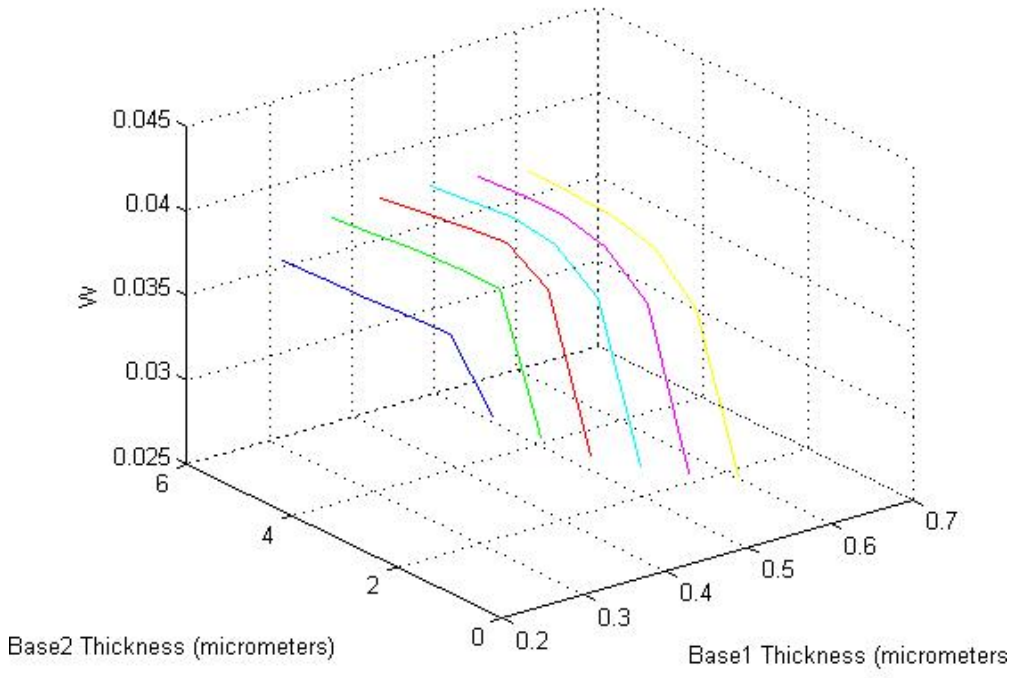

Figure 45. Iteration test for different bases thicknesses at 325K

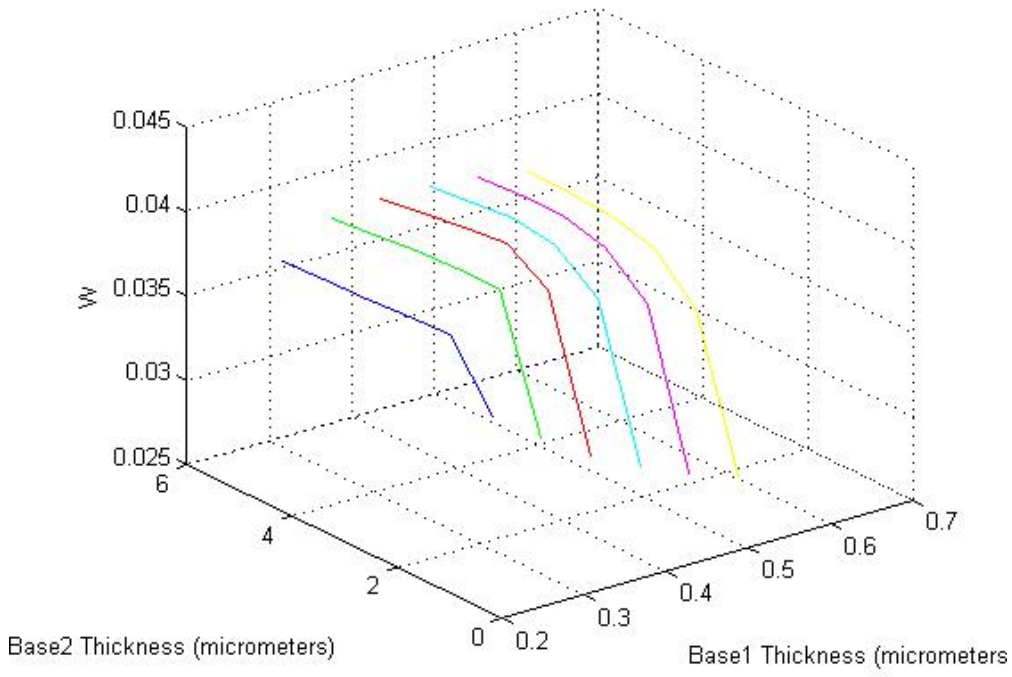

Figure 46. Iteration test for different bases thicknesses at 350K

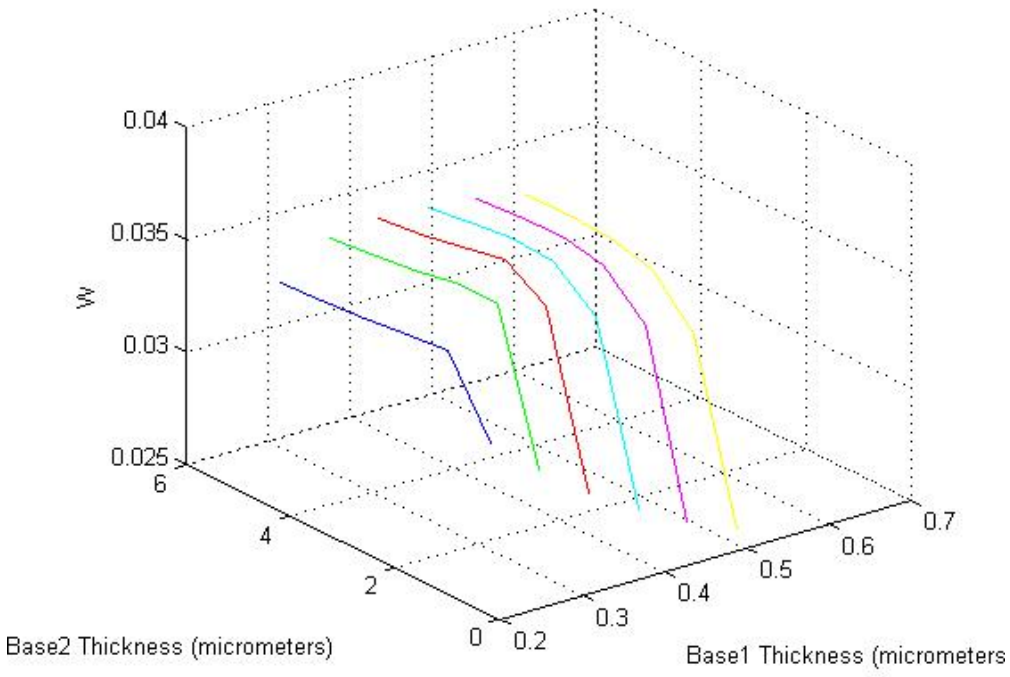

Figure 47. Iteration test for different bases thicknesses at 375K

Temperature (K)	InGaP Base Thickness (microns)	GaAs Base Thickness (microns)	Max Power (W)	Efficiency (%)
300K				
	0.22	1.3	39.65892388	29.31184
	0.28	2.1	41.51190148	30.68138
	0.34	2.1	42.05842737	31.08531
	0.4	2.9	41.51101768	30.68072
	0.46	3.7	40.83436732	30.18061
	0.52	3.7	40.14097529	29.66813
325K				
	0.22	1.3	37.56438997	27.76377677
	0.28	2.1	39.19973794	28.97245968
	0.34	2.1	39.66902577	29.31930951
	0.4	2.9	39.13769412	28.92660319
	0.46	3.7	38.49350299	28.45048262
	0.52	3.7	37.85812379	27.98087494
350K				
	0.22	1.3	35.06466115	25.91623145
	0.28	1.3	36.49009365	26.96976619
	0.34	2.1	36.84558054	27.23250594
	0.4	2.9	36.3107759	26.83723274
	0.46	2.9	35.75063575	26.42323411
	0.52	3.7	35.16289268	25.98883421
375K				
	0.22	1.3	32.73228743	24.19237799
	0.28	1.3	33.97532835	25.11110743
	0.34	2.1	34.21425425	25.28769716
	0.4	2.9	33.72567945	24.92659235
	0.46	2.9	33.21896504	24.55208059
	0.52	3.7	32.66534961	24.14290437

Table 4. Results of six iteration test of changing base thicknesses.

The highlighted P_{max} results for Table 4 were plotted versus temperature to graphically display the optimal combinations (see Figure 48). The optimal InGaP base thickness was always kept at 0.34 μm so only the GaAs thickness was plotted for comparison. The best power, regardless of temperature, was found at 0.34 and 2.1 μm for the InGaP and GaAs base thicknesses.

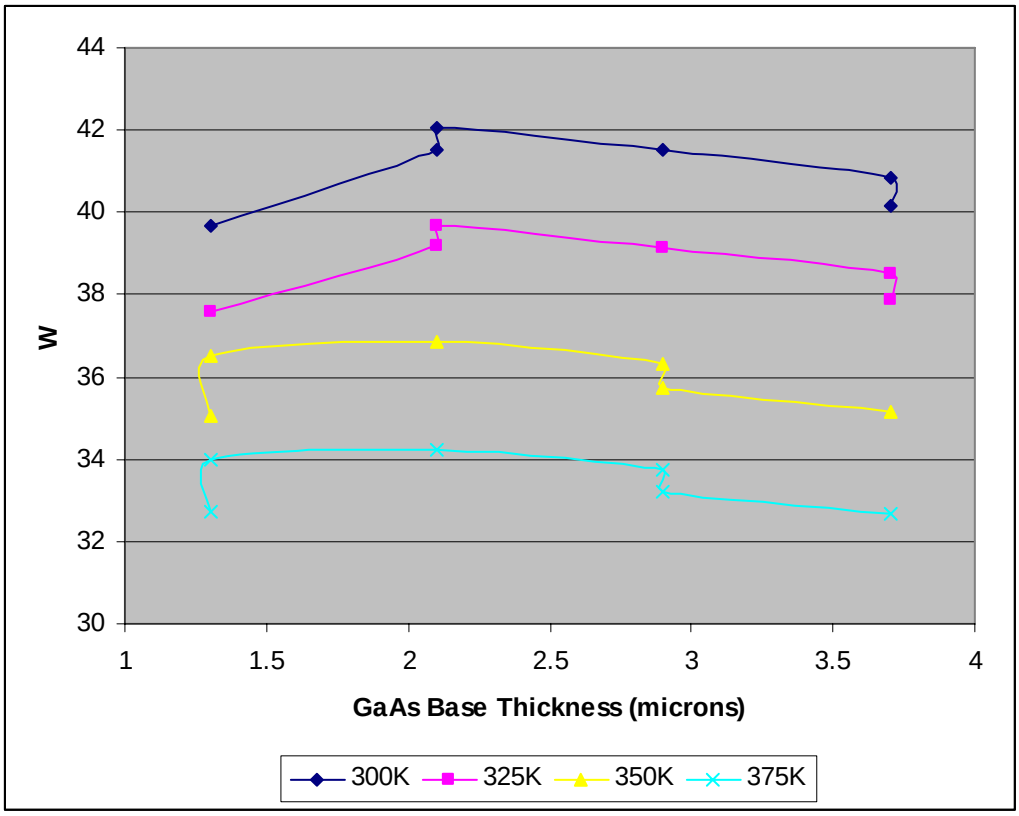

Figure 48. Optimal Pmax for a triple junction cell at various temperatures and GaAs base thickness.

The next step was to use a finer iteration steps to further test around the expected values for design changes. The bases varied from 0.26 to 0.4 μm for the InGaP base to 1.3 to 2.6 μm for the GaAs base. Due to the results of the previous test, only the extreme temperatures were run. The results of varying the temperature produced a cell that again seemed to have the same dimensions as the baseline cell (see Figures 49 and 51). A two dimensional representation was generated to more clearly visualize the peak efficiency (see Figures 50 and 52).

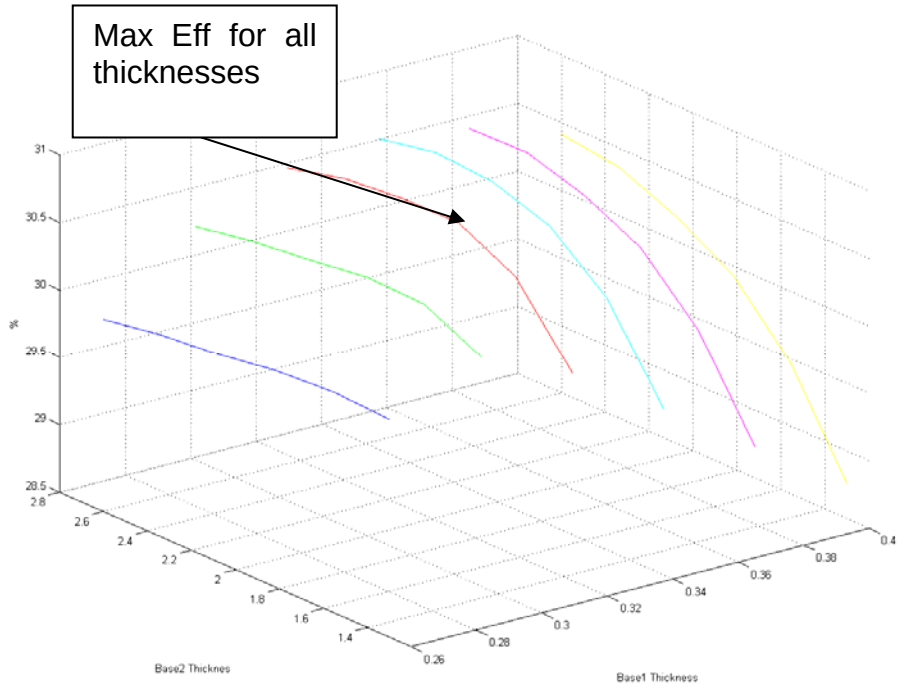

Figure 49. Efficiency in triple junction cell with variations in thickness of two bases at 300K.

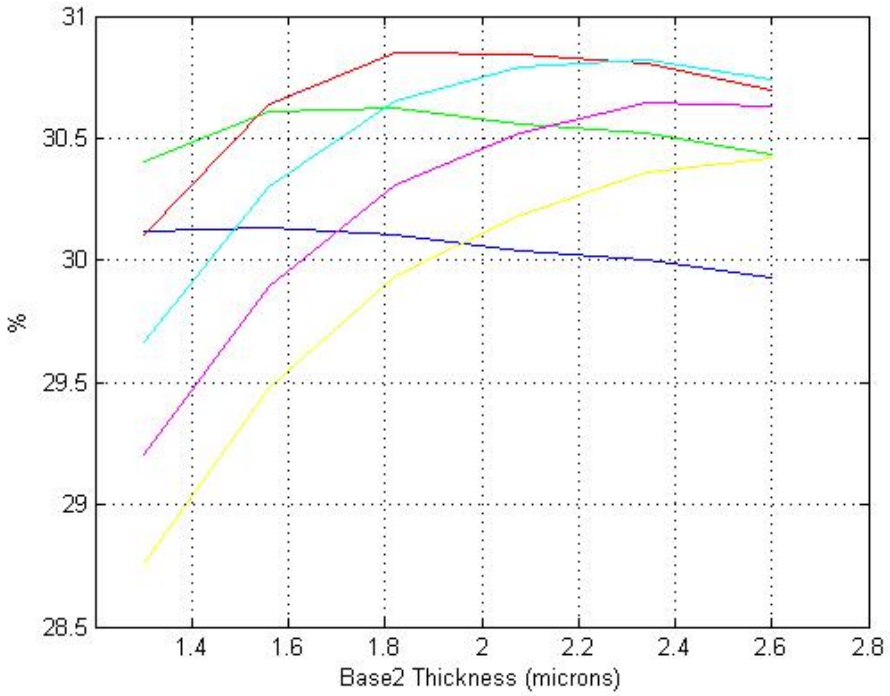

Figure 50. 2D representation of Figure 49 to show base thickness maximum efficiency point at 300K.

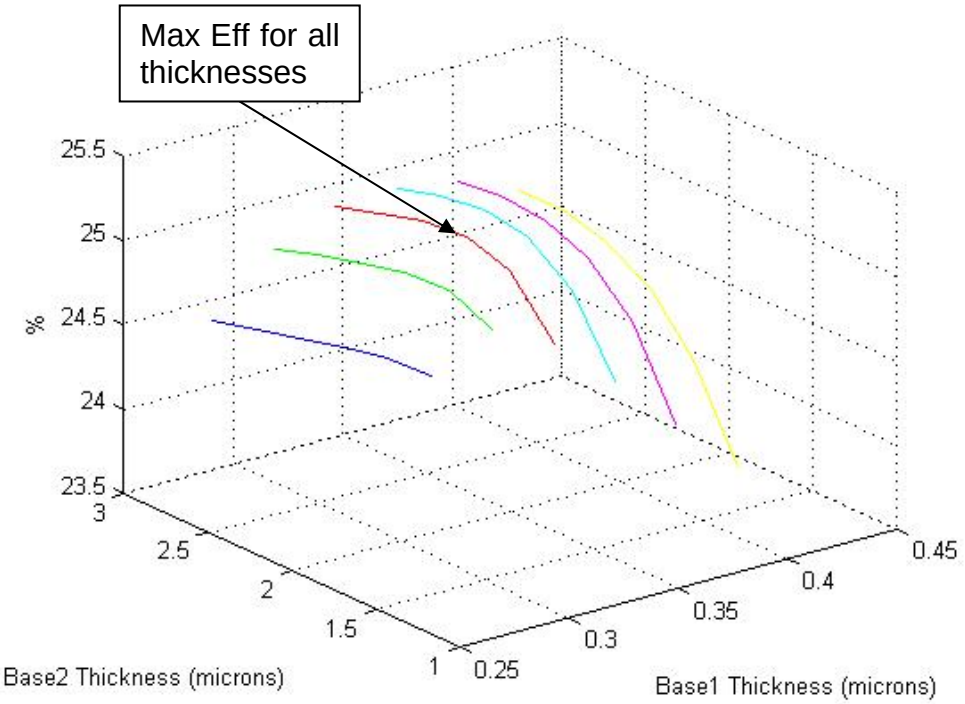

Figure 51. Efficiency in triple junction cell with variations in thickness of two bases at 375K.

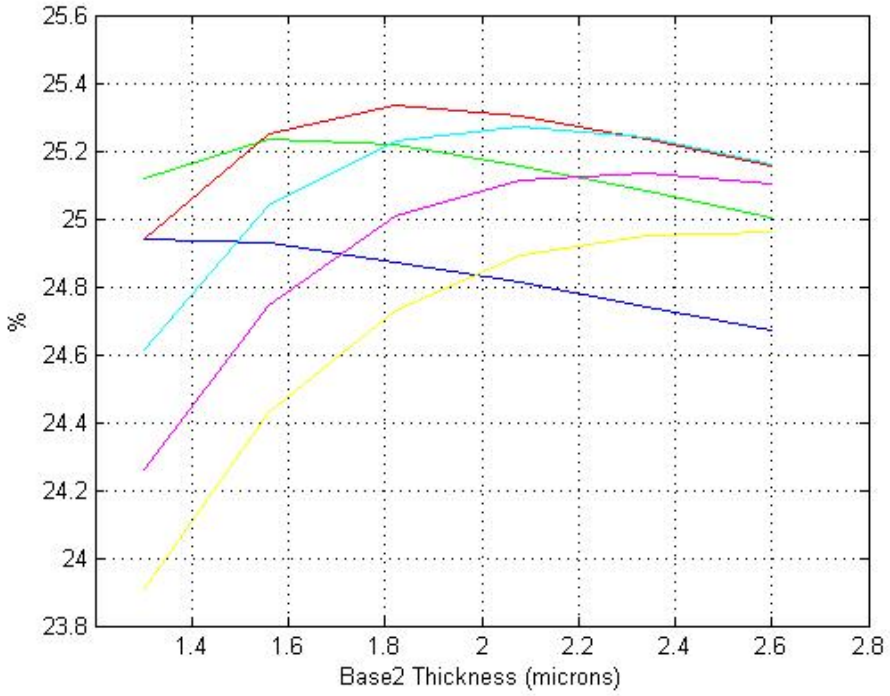

Figure 52. 2D representation of Figure 51 to show base thickness maximum efficiency point at 375K.

These results are in line with premise that the decrease in band gap energy due to an increase in temperature is the driving factor for the reduction in efficiency with increasing temperature. The thickness of the bases of the cell affects current and shadowing but would have little effect on band gap energy, thus little effect on the overall voltage. Based on [3] a triple junction cell with better current matching at higher temperatures could produce a more efficient cell but, due to time constraints and the restriction on varying only two parameters, this current matched cell was not realized.

An interesting note was that a different cell was realized from Bates' original cell. A thickness of 0.316 μm and 1.82 μm for InGaP and GaAs respectively achieved a P_{max} of 41.7388mW/cm^2, compared to the original P_{max} of 41.7512 mW/cm^2. This cell is very similar in power, even though the base thicknesses are about 0.3 and 1.2 μm different from the original. From the simulation data, the P_{max} at the original base thicknesses was 41.6575 mW/cm^2. The *I-V* curves for the different cells are in Figures 38 and 53.

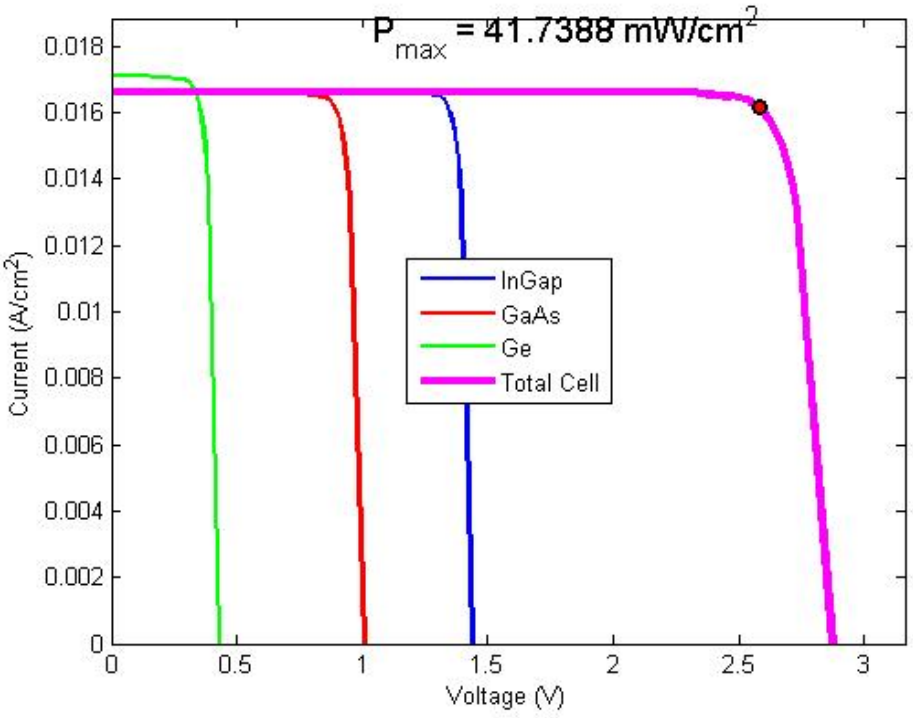

Figure 53. *I-V* curve based on new bases discovered in smaller iteration test at 300K.

This discrepancy seems to be a limitation of the simulation. The simulation was able to obtain accuracy at about 0.22% when the power at the same base thicknesses is compared. Another source of error is the simplistic optimization method. In the new cell, the currents are mismatched which could lead to some power loss for that base combination. In turn, a cell with slight differences could produce more power, giving the appearance that a new cell is optimum.

With varying base thicknesses having little effect on the solar cell, doping concentrations were examined. The premise that the temperature variation of the band gap energy could provide a more optimal solar cell was investigated by varying the doping levels of the top junctions. The extreme temperature limits were examined for variation. The result was less than envisioned.

The doping variation near Bates' optimal configuration was extremely close that small error could sway the results. The base doping concentrations for InGaP and GaAs was varied by 1e10 for 300K. Table 5 shows how the results varied so slightly. The maximum power appears to be at a different doping level than Bates' original levels of 1.50e17 (highlighted in orange). The new max power point is at 1.50e17 for the InGaP but the GaAs shifts to 1.50e14 (highlighted in yellow). This change is about a 4.32% increase in power. As with the slight increase in power obtained by shrinking the GaAs thickness, this small shift in doping concentration has improved upon Bates' original solar cell. Bates' optimization technique only varied the InGaP layer with the assumption that the single junction GaAs cell previously optimized would be the best solution for the multi-junction solar cell. This simulation discovered a different optimization by varying the GaAs junction of the multi-junction cell.

InGaP Base doping (cm^-3)	GaAs Base doping (cm^-3)	Max Power (W)
at 300K		
1.50E+14	1.50E+14	42.1002
1.50E+14	1.50E+15	41.9993
1.50E+14	1.50E+16	41.7372
1.50E+14	1.50E+17	40.3547
1.50E+14	1.50E+18	29.9127
1.50E+15	1.50E+14	42.1004
1.50E+15	1.50E+15	41.9995
1.50E+15	1.50E+16	41.7374
1.50E+15	1.50E+17	40.3549
1.50E+15	1.50E+18	29.9128
1.50E+16	1.50E+14	42.1014
1.50E+16	1.50E+15	42.0005
1.50E+16	1.50E+16	41.7384
1.50E+16	1.50E+17	40.3558
1.50E+16	1.50E+18	29.9132
1.50E+17	1.50E+14	42.1055
1.50E+17	1.50E+15	42.0046
1.50E+17	1.50E+16	41.7425
1.50E+17	1.50E+17	40.3602
1.50E+17	1.50E+18	29.9157
1.50E+18	1.50E+14	42.1008
1.50E+18	1.50E+15	42.0000
1.50E+18	1.50E+16	41.7417
1.50E+18	1.50E+17	40.3676
1.50E+18	1.50E+18	29.9233

Table 5. Results of varying bases doping concentrations for a triple junction solar cell at 300K with max power in yellow and expected value in orange.

Next the same cell was tested at 375K. At first the results seemed to indicate a change in doping created a shift in peak power. The maximum power at 375K changed to 1.5e16 base doping for InGaP and 1.5e14 for the GaAs base doping (see Table 6). This shift is so slight that it cannot be treated as better design. The percent difference in max power is less than 0.008%, which could be attributed to simulation error. But a comparison to Bates' original cell parameters, an increase by 3.5% is realized at the higher temperature.

InGaP Base doping (cm^-3)	GaAs Base doping (cm^-3)	Max Power (W)
at 375K		
1.50E+14	1.50E+14	34.4739
1.50E+14	1.50E+15	34.4191
1.50E+14	1.50E+16	34.2763
1.50E+14	1.50E+17	33.3120
1.50E+14	1.50E+18	24.8739
1.50E+15	1.50E+14	34.4740
1.50E+15	1.50E+15	34.4192
1.50E+15	1.50E+16	34.2764
1.50E+15	1.50E+17	33.3120
1.50E+15	1.50E+18	24.8739
1.50E+16	1.50E+14	34.4739
1.50E+16	1.50E+15	34.4191
1.50E+16	1.50E+16	34.2762
1.50E+16	1.50E+17	33.3119
1.50E+16	1.50E+18	24.8737
1.50E+17	1.50E+14	34.4712
1.50E+17	1.50E+15	34.4164
1.50E+17	1.50E+16	34.2738
1.50E+17	1.50E+17	33.3101
1.50E+17	1.50E+18	24.8733
1.50E+18	1.50E+14	34.4411
1.50E+18	1.50E+15	34.3864
1.50E+18	1.50E+16	34.2458
1.50E+18	1.50E+17	33.2940
1.50E+18	1.50E+18	24.8727

Table 6. Results of varying bases doping concentrations for a triple junction solar cell at 375K with max power in yellow, expected value in orange, and Bates' original value in turquoise.

The conclusion was that changes in the base thicknesses and doping levels do not further enhance the optimized solar cell as temperature increases.

Finally, the emitter doping levels for the InGaP and GaAs were varied to observe the changes as temperature rose to 375K. Table 7 shows the baseline maximum power at 300K as the doping changes. The result is a slight improvement over Bates' original max power, a 0.51% increase in power by changing the GaAs emitter doping concentration to 2.0e15 cm^{-3}.

InGaP Emitter doping (cm^-3)	GaAs Emitter doping (cm^-3)	Max Power (W)
at 300K		
2.00E+15	2.00E+15	41.9417
2.00E+15	2.00E+16	41.9403
2.00E+15	2.00E+17	41.9220
2.00E+15	2.00E+18	41.7232
2.00E+15	2.00E+19	40.5385
2.00E+15	2.00E+20	37.1817
2.00E+16	2.00E+15	41.9420
2.00E+16	2.00E+16	41.9406
2.00E+16	2.00E+17	41.9223
2.00E+16	2.00E+18	41.7235
2.00E+16	2.00E+19	40.5388
2.00E+16	2.00E+20	37.1819
2.00E+17	2.00E+15	41.9456
2.00E+17	2.00E+16	41.9442
2.00E+17	2.00E+17	41.9260
2.00E+17	2.00E+18	41.7272
2.00E+17	2.00E+19	40.5422
2.00E+17	2.00E+20	37.1848
2.00E+18	2.00E+15	41.9566
2.00E+18	2.00E+16	41.9551
2.00E+18	2.00E+17	41.9369
2.00E+18	2.00E+18	41.7388
2.00E+18	2.00E+19	40.5538
2.00E+18	2.00E+20	37.1946
2.00E+19	2.00E+15	41.9350
2.00E+19	2.00E+16	41.9336
2.00E+19	2.00E+17	41.9152
2.00E+19	2.00E+18	41.7259
2.00E+19	2.00E+19	40.5501
2.00E+19	2.00E+20	37.1959
2.00E+20	2.00E+15	41.6576
2.00E+20	2.00E+16	41.6566
2.00E+20	2.00E+17	41.6440
2.00E+20	2.00E+18	41.4876
2.00E+20	2.00E+19	40.4700
2.00E+20	2.00E+20	37.1665

Table 7. Results of varying emitter doping concentrations for a triple junction solar cell at 300K with max power in yellow and expected value in orange.

The results of doping variation for the emitters at 375K are shown in Table 8.

InGaP Emitter doping (cm^-3)	GaAs Emitter doping (cm^-3)	Max Power (W)
at 375K		
2.00E+15	2.00E+15	34.4103
2.00E+15	2.00E+16	34.4092
2.00E+15	2.00E+17	34.3958
2.00E+15	2.00E+18	34.2544
2.00E+15	2.00E+19	33.1902
2.00E+15	2.00E+20	30.5314
2.00E+16	2.00E+15	34.4108
2.00E+16	2.00E+16	34.4097
2.00E+16	2.00E+17	34.3962
2.00E+16	2.00E+18	34.2548
2.00E+16	2.00E+19	33.1907
2.00E+16	2.00E+20	30.5317
2.00E+17	2.00E+15	34.4159
2.00E+17	2.00E+16	34.4148
2.00E+17	2.00E+17	34.4014
2.00E+17	2.00E+18	34.2598
2.00E+17	2.00E+19	33.1953
2.00E+17	2.00E+20	30.5356
2.00E+18	2.00E+15	34.4324
2.00E+18	2.00E+16	34.4313
2.00E+18	2.00E+17	34.4179
2.00E+18	2.00E+18	34.2762
2.00E+18	2.00E+19	33.2112
2.00E+18	2.00E+20	30.5489
2.00E+19	2.00E+15	34.4174
2.00E+19	2.00E+16	34.4163
2.00E+19	2.00E+17	34.4028
2.00E+19	2.00E+18	34.2645
2.00E+19	2.00E+19	33.2095
2.00E+19	2.00E+20	30.5507
2.00E+20	2.00E+15	34.2091
2.00E+20	2.00E+16	34.2083
2.00E+20	2.00E+17	34.1992
2.00E+20	2.00E+18	34.0686
2.00E+20	2.00E+19	33.1163
2.00E+20	2.00E+20	30.5182

Table 8. Results of varying emitter doping concentrations for a triple junction solar cell at 375K with max power equaling the expected value in yellow, and Bates' original value in turquoise.

The increase in temperature had no apparent effect on the optimum cell design. The slight improvement over the original design held at 0.45%. The

conclusion is that temperature variations coupled with the emitter doping concentrations do not have an appreciable effect on the solar cell design for temperature ranges from 300K to 375K.

As stated previously the driving factor for the drop in power or efficiency was the change in band gap energy. The band gap energy for GaAs is shown in Figure 54. The calculated values are given by:

$$E_g = \frac{1.515 - 5.5 \bullet 10^{-4} T^2}{T + 255} eV$$

For the range of 300K to 375K the calculated change is only about 0.033eV. Such a small change in band gap energy may not appreciably change the triple junction cell output.

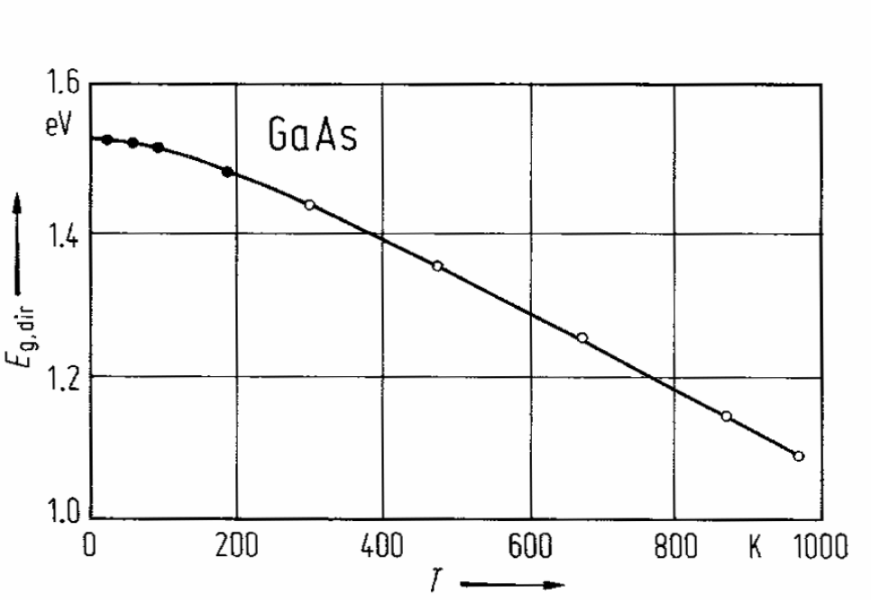

Figure 54. Direct energy gap for GaAs as a function of temperature [from 26].

The other temperature effect issue depends on photogeneration and recombination. In other words the number of free carriers available to conduct electricity affects the power output. Again, at this limited temperature range, the majority carriers do not vary enough for an appreciable effect. As stated in [12] GaAs does not deviate from its extrinsic properties until an operating temperature

around 700K. This means that a material's carrier concentrations, thus its thermal generation and recombination, do not differ greatly for the temperature ranges involved. These observations lead to some conclusions about the capability of the simulations and the nature of solar cells.

Though most temperature data of operating solar cells is proprietary, a typical operating temperature of 60 ^{o}C or 333K was supplied by a Space Systems, Loral engineer [31]. For this reason the simulation temperatures stopped at 375K, in order to cover these typical temperatures encountered by space craft.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The use of simulation software has many diverse possibilities. The precious work using Silvaco simulation tools has been repeatedly demonstrated as a powerful tool to better understand the complexities of the multi-junction solar cell. Without the need to design, build, and test a practical solar cell proposal, the photovoltaic industry can simulate a solar cell before fabrication, saving time and money. The true power of the simulation is the ability to optimize a multi-junction solar cell with all of its variable parameters while taking into account external forces, such as radiation or temperature.

The first step in this thesis was to recreate parts of previous thesis work to demonstrate the feasibility of the software. A single junction cell was used as an example to show the temperature effects on solar cells. Next, the temperature effects on a triple junction cell were then simulated. The results were compared to measured and calculated data. With this confidence, the optimization of a triple junction solar cell was attempted.

The optimization of a triple junction cell was daunting. With only the programming ability to vary two parameters at a time, many attempts to improve a cell at higher temperature was conducted. The result was a minimal improvement that could just be within the simulation error. Alteration over broader range of parameter would give a more definitive answer. Also, the constraint to use realistic operating temperatures may be too narrow to demonstrate an optimal design. Due to these limitations, slight improvements were made over the original cell design but no temperature dependent improvements were achieved.

The tests presented in this thesis have shown that Silvaco ATLAS can be used to accurately simulate temperature changes in single and multi-junction solar cells. This simulation can then be used as a valuable tool to better predict environmental effects on solar cells. This software may be a valuable asset for a design outside of the typical Earth operating environment, such as a Mercury mission or the Solar Probe.

B. RECOMMENDATIONS

The first recommendation is to development a better optimization program, such as the one Bates used. Only changing two parameters at a time is very limiting to the overall design and can lead to false conclusions. Bates' genetic algorithm showed a way to converge to a solution while vary many parameters. The Taguchi statistical method was reviewed for use in identifying parameters and interactions, but was not implemented due to time constraints. The complexity involved in the optimization of the triple junction cell needs improvement.

A second recommendation is to investigate other areas that are effected by temperature changes, such as the impurity ionization. The default settings were used in this model but Silvaco permits different impurities to be input into the simulated solar cell. Temperature affects the concentration of these ionized materials and could be used to develop a more efficient solar cell at a higher temperature.

This thesis approached the topic as an electrical solution. Other solutions could help solve this loss in solar cell efficiency due to heating. The obvious one is to lower the solar arrays temperature to permit the solar cells to operating at an optimal temperature. Cooling the arrays could be active, like with a cryogenic or passive, such as off point to maintain a more optimal temperature. Each has drawbacks as with many engineering trades. Another solution could be the

material itself. As material engineers develop new processes and materials for solar manufacturing, Silvaco software can simulate these new substances at various environmental conditions.

Lastly, as solar concentrators become more prominent for both terrestrial and space applications, the heating of the solar cells by the concentrators can be simulated in Silvaco. Not only the effects on the solar cells can be predicted through the simulations, but an optimum overarching design could be realized. A design could change solar cell parameters while taking into account different input light spectrum, along with the best operating temperature for the assembly. The power of the software simulations comes from the ability to program the different inputs and external factors, thus deriving the optimal solar cell device.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. ATLAS SOURCE CODE

Silvaco DeckBuild input file for various single junction solar cell configurations. Basic file was taken from Bates' thesis [3], and then modified slightly for temperature variation.

A. SINGLE JUNCTION SOLAR CELL (EITHER INGAP, GAAS, OR GE)

```
go atlas

######## Single Junction Cell with set up for GaAs, InGaP, and Ge parameters
########
### from Bates

### Constant Definitions
set cellWidth=5.000000e+002
set capWidthpercent=8.000000e+000
set divs=1.000000e+001
set contThick=1.000000e-001
set capThick=3.000000e-001
set capDop=1.000000e+020
set windowThick=0.01
set winDop=2.15e17
set emitterThick=0.01
set emitDop=1e16
set baseThick=3.19467
set baseDop=1e16
set bsfThick=0.03533
set bsfDop=2.15e19

set cellWidthDiv=$cellWidth/$divs
set width3d=100e6/$cellWidth
set capWidth=0.01*$capWidthpercent*$cellWidth/2
set capWidthDiv=$capWidth/($divs/2)
set cellWidthHalf=$cellWidth/2

### Building the cell from the bottom up
set bsfLo=0
set bsfHi=$bsfLo-$bsfThick
set bsfDiv=$bsfThick/$divs

set baseLo=$bsfHi
set baseHi=$baseLo-$baseThick
set baseMid=$baseLo-$baseThick/2
set baseDiv=$baseThick/$divs

set emitterLo=$baseHi
set emitterHi=$emitterLo-$emitterThick
set emitterDiv=$emitterThick/$divs

set windowLo=$emitterHi
set windowHi=$windowLo-$windowThick
set windowDiv=$windowThick/$divs

set capLo=$windowHi
```

```
set capHi=$capLo-$capThick

set contLo=$capHi
set contHi=$contLo-$contThick
set contDiv=$contThick/$divs

set lightY=$emitterHi-5

### Defining the Mesh
mesh width=$width3d
## X-Mesh
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv
x.mesh loc=-$capWidth spac=$capWidthDiv
x.mesh loc=$capWidth spac=$capWidthDiv
x.mesh loc=$cellWidthHalf spac=$cellWidthDiv

## Y-Mesh
# Top contact
y.mesh loc=$contHi spac=0
y.mesh loc=$contLo spac=0
# Cap
# Window
y.mesh loc=$windowHi spac=$windowDiv
y.mesh loc=$windowLo spac=$windowDiv
# Emitter
y.mesh loc=$emitterLo spac=$emitterDiv
# Base
y.mesh loc=$baseMid spac=$baseDiv
# BSF
y.mesh loc=$bsfHi spac=$bsfDiv
y.mesh loc=$bsfLo spac=$bsfDiv

#######################################
## CURRENTLY SET UP FOR:  GaAs CELL ##
#######################################

### Regions [for InGaP cell, change region 1 to GaAs (v. Vacuum) and remove
region 8 (bogus contact)]
##         [for all others, change materials only]

# Cap
region num=8 material=Vacuum x.min=-$capWidth x.max=$capWidth y.min=$contHi
y.max=$contLo
region num=1 material=Vacuum x.min=-$capWidth x.max=$capWidth y.min=$capHi
y.max=$capLo
region num=2 material=Vacuum x.min=-$cellWidthHalf x.max=-$capWidth
y.min=$contHi y.max=$capLo
region num=3 material=Vacuum x.min=$capWidth x.max=$cellWidthHalf y.min=$contHi
y.max=$capLo
# Window [for Ge cell, use AlGaAs with x.comp=0.2]
region num=4 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi y.max=$windowLo
#region num=4 material=AlGaAs x.comp=0.2 x.min=-$cellWidthHalf
x.max=$cellWidthHalf y.min=$windowHi y.max=$windowLo
# Emitter
region num=5 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$emitterHi y.max=$emitterLo
# Base
region num=6 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$baseHi y.max=$baseLo
# BSF
```

```
region num=7 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$bsfHi y.max=$bsfLo

### Electrodes [for InGaP cell, add cathode (gold) and remove cathode
(conductor)]

#electrode name=cathode material=Gold x.min=-$capWidth x.max=$capWidth
y.min=$contHi y.max=$contLo
electrode name=cathode x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi y.max=$windowHi
electrode name=anode x.min=-$cellWidthHalf x.max=$cellWidthHalf y.min=$bsfLo
y.max=$bsfLo

### Doping [for InGaP cell, uncomment cap doping]

# Cap
#doping uniform region=1 n.type conc=$capDop
# Window
doping uniform region=4 n.type conc=$winDop
# Emitter
doping uniform region=5 n.type conc=$emitDop
# Base
doping uniform region=6 p.type conc=$baseDop
# BSF
doping uniform region=7 p.type conc=$bsfDop

### Material properties  ### not all properties needed for all cells ###

# Opaque contact [comment out for InGaP cell]
material region=8 real.index=1.2 imag.index=1.8
# Vacuum (for zero reflection) [change to match window material (InGaP use
Vacuum_AlInP)]
#                              [for InGaP cell, comment out region 1]
material region=1 index.file=Vacuum_InGaP.opt
material region=2 index.file=Vacuum_InGaP.opt
material region=3 index.file=Vacuum_InGaP.opt
# GaAs
material material=GaAs EG300=1.424 PERMITTI-VITY=12.9 AFFINITY=4.07 \
  NC300=4.7E17 NV300=9E18 INDEX.FILE=GaAs.opt COPT=7.2E-10 \
  AUGN=5E-30 AUGP=1E-31
# InGaP
material material=InGaP EG300=1.9 PERMITTI-VITY=11.62 AFFINITY=4.16 \
  NC300=1.3E20 NV300=1.28E19 index.file=InGaP.opt COPT=1E-10 \
  MUN=4000 MUP=200 AUGN=3e-30 AUGP=3E-30
# Ge
material material=Ge EG300=0.661 PERMITTI-VITY=16.2 AFFINITY=4 \
  NC300=1E19 NV300=5E18 index.file=Ge.opt COPT=6.41E-14 \
  MUN=3900 MUP=1900 AUGN=1E-30 AUGP=1E-30
# AlGaAs
material material=AlGaAs MUN=9000 MUP=100 INDEX.FILE=AlGaAs.opt
# AlInP (=InAsP)
material material=InAsP EG300=2.4 PERMITTI-VITY=11.7 AFFINITY=4.2 \
  NC300=1.08E20 NV300=1.28E19 index.file=AlInP.opt COPT=1.2E-10 \
  MUN=2291 MUP=142 AUGN=9E-31 AUGP=9E-31
# AlInGaP (=InAlAsP)
material material=InAlAsP EG300=2.4 PERMITTI-VITY=11.7 AFFINITY=4.2 \
  NC300=1.2E20 NV300=1.28E19 index.file=AlInP.opt COPT=1E-10 \
  MUN=2150 MUP=141 AUGN=3e-30 AUGP=3E-30
# InGaNAs
material material=InGaNAs EG300=1.0 PERMITTI-VITY=11.7 AFFINITY=4.05 \
  NC300=3.2e19 NV300=1.8e19 index.file=InGaNAs.opt COPT=7.2e-10 \
  MUN=3000 MUP=150
```

```
# Gold
material material=Gold real.index=1.2 imag.index=1.8

### Models [InGaP cell, 1; GaAs cell, 5&6; InGaNAs cell, 7] ###
### if no temperature specified then default temperature=300K ###

#models region=1 CONMOB temp=340 print
models region=5 CONMOB temp=340 print
models region=6 CONMOB temp=340 print
#models region=7 CONMOB temp=340 print
models OPTR BGN temp=340 print

### Light beams [GaAs b1,0.55-0.75,200 b2,0.75-0.88,65] 0.12-2.7,50 [630,825]

### original beams ### found that current was not always consistent with
expected values
### so switched to higher resolution multi-junction cell light beam ###

#beam num=1 x.origin=0 y.origin=$lightY angle=90 back.refl
power.file=AM0nrel.spec \
#  wavel.start=0.55 wavel.end=0.75 wavel.num=200
#beam num=2 x.origin=0 y.origin=$lightY angle=90 back.refl
power.file=AM0nrel.spec \
#  wavel.start=0.75 wavel.end=0.88 wavel.num=65

##### from Bates mj cell #####
beam num=1 x.origin=0 y.origin=$lightY angle=90 back.refl
power.file=AM0nrel.spec \
  wavel.start=0.12 wavel.end=0.75 wavel.num=630
beam num=2 x.origin=0 y.origin=$lightY angle=90 back.refl
power.file=AM0nrel.spec \
  wavel.start=0.7501 wavel.end=2.4 wavel.num=825

### develops, saves, and plots structure file for review, can omit once
### satisfied structure is achieved
s outfile=SingleCell_webf.str
tonyplot SingleCell_webf.str

### Start solution set
solve init
method gummel newton maxtraps=10 itlimit=25
solve b1=0.9, b2=0.9

## Getting Isc for I-V curve points
method newton maxtraps=10 itlimit=100
## beam is set to 90% to take into account reflection losses
solve b1=0.9, b2=0.9
extract name="isc" max(i."cathode")
set isc=$isc*$width3d
set i1=$isc/10
set i2=$i1+$isc/10
set i3=$i2+$isc/10
set i4=$i3+$isc/10
set i5=$i4+$isc/10
set i6=$i5+$isc/20
set i7=$i6+$isc/20
set i8=$i7+$isc/20
set i9=$i8+$isc/20
set i10=$i9+$isc/20
set i11=$i10+$isc/40
set i12=$i11+$isc/40
set i13=$i12+$isc/40
```

```
set i14=$i13+$isc/40
set i15=$i14+$isc/40
set i16=$i15+$isc/80
set i17=$i16+$isc/80
set i18=$i17+$isc/80
set i19=$i18+$isc/80
set i20=$i19+$isc/80
set i21=$i20+$isc/80
set i22=$i21+$isc/80
set i23=$i22+$isc/80
set i24=$i23+$isc/80
set i25=$i24+$isc/80-0.00001
##

log outfile=GaAs_300K.log

method newton maxtraps=10 itlimit=100
solve b1=0.9, b2=0.9

contact name=anode current
method newton maxtraps=10 itlimit=100

## Pmax points [InGaP 18-25; GaAs 15-25; InGaNAs 13-25; Ge 11-25]
solve ianode=-$i25 b1=0.9, b2=0.9
solve ianode=-$i24 b1=0.9, b2=0.9
solve ianode=-$i23 b1=0.9, b2=0.9
solve ianode=-$i22 b1=0.9, b2=0.9
solve ianode=-$i21 b1=0.9, b2=0.9
solve ianode=-$i20 b1=0.9, b2=0.9
solve ianode=-$i19 b1=0.9, b2=0.9
solve ianode=-$i18 b1=0.9, b2=0.9
solve ianode=-$i17 b1=0.9, b2=0.9
solve ianode=-$i16 b1=0.9, b2=0.9
solve ianode=-$i15 b1=0.9, b2=0.9
solve ianode=-$i14 b1=0.9, b2=0.9
#solve ianode=-$i13 b1=0.9, b2=0.9
#solve ianode=-$i12 b1=0.9, b2=0.9
#solve ianode=-$i11 b1=0.9, b2=0.9
#solve ianode=-$i10 b1=0.9, b2=0.9
#solve ianode=-$i9 b1=0.9, b2=0.9
#solve ianode=-$i8 b1=0.9, b2=0.9
#solve ianode=-$i7 b1=0.9, b2=0.9
#solve ianode=-$i6 b1=0.9, b2=0.9
#solve ianode=-$i5 b1=0.9, b2=0.9
#solve ianode=-$i4 b1=0.9, b2=0.9
#solve ianode=-$i3 b1=0.9, b2=0.9
#solve ianode=-$i2 b1=0.9, b2=0.9
#solve ianode=-$i1 b1=0.9, b2=0.9
##

solve ianode=0 b1=0.9, b2=0.9

log off

## Full I-V curve plot
tonyplot GaAs_300K.log -set pmax.set
##
```

B. TRIPLE JUNCTION SOLAR CELL (INGAP/GAAS/GE)

Silvaco DeckBuild input file for various triple junction solar cell configurations. Basic file was taken from Bates' thesis [3]., then modified slightly for temperature variation.

```
go atlas
set cellWidth=5.000000e+002
set capWidthpercent=8.000000e+000
set divs=1.000000e+001
set tunnelThick=3.000000e-001
set contThick=1.000000e-001
set capThick=3.000000e-001
set capDop=1.950000e+018
set windowThick1=3.000000e-002
set winDop1=1.950000e+018
### in the constant definitions, the parameter was set to the
### first iteration number called for by filerw.m
### in this set up the emitDop1 & 2 level is being iterated so
### emitDop1temp was set to the first value of a in inter_test_mj.m
set emitterThick1=5.000000e-002
### original value ###
###set emitDop1=2.000000e+018
set emitDop1temp=1
set emitDop1=(10^$emitDop1temp)*2e+014
set baseThick1=0.341
set baseDop1=1.50000000000000000e+017
set bsfThick1=3.000000e-002
set bsfDop1=2.000000e+018
set windowThick2=5.000000e-002
set winDop2=1.000000e+019
set emitterThick2=1.000000e-001
### original value ###
###set emitDop2=2.000000e+018
set emitDop2temp=3
set emitDop2=(10^$emitDop2temp)*2e+014
set baseThick2=2.1
set baseDop2=1.000000e+017
set bsfThick2=1.000000e-001
set bsfDop2=2.000000e+018
set windowThick3=5.000000e-002
set winDop3=7.000000e+018
set emitterThick3=1.000000e-001
set emitterThick3=0.05
set emitDop3=2.000000e+018
set baseThick3=3.000000e+002
set baseThick3=220
set baseDop3=1.000000e+017


set cellWidthDiv=$cellWidth/$divs
set width3d=100e6/$cellWidth
set capWidth=0.01*$capWidthpercent*$cellWidth/2
set capWidthDiv=$capWidth/($divs/2)
set cellWidthHalf=$cellWidth/2

set baseLo3=0
set baseHi3=$baseLo3-$baseThick3
```

```
set baseMid3=$baseLo3-$baseThick3/2
set baseDiv3=$baseThick3/$divs

set emitterLo3=$baseHi3
set emitterHi3=$emitterLo3-$emitterThick3
set emitterDiv3=$emitterThick3/$divs

set windowLo3=$emitterHi3
set windowHi3=$windowLo3-$windowThick3
set windowDiv3=$windowThick3/$divs

set midTunnel23=$windowHi3-$tunnelThick/2

set bsfLo2=$windowHi3-$tunnelThick
set bsfHi2=$bsfLo2-$bsfThick2
set bsfDiv2=$bsfThick2/$divs

set baseLo2=$bsfHi2
set baseHi2=$baseLo2-$baseThick2
set baseMid2=$baseLo2-$baseThick2/2
set baseDiv2=$baseThick2/$divs

set emitterLo2=$baseHi2
set emitterHi2=$emitterLo2-$emitterThick2
set emitterDiv2=$emitterThick2/$divs

set windowLo2=$emitterHi2
set windowHi2=$windowLo2-$windowThick2
set windowDiv2=$windowThick2/$divs

set midTunnel12=$windowHi2-$tunnelThick/2

set bsfLo1=$windowHi2-$tunnelThick
set bsfHi1=$bsfLo1-$bsfThick1
set bsfDiv1=$bsfThick1/$divs

set baseLo1=$bsfHi1
set baseHi1=$baseLo1-$baseThick1
set baseMid1=$baseLo1-$baseThick1/2
set baseDiv1=$baseThick1/$divs

set emitterLo1=$baseHi1
set emitterHi1=$emitterLo1-$emitterThick1
set emitterDiv1=$emitterThick1/$divs

set windowLo1=$emitterHi1
set windowHi1=$windowLo1-$windowThick1
set windowDiv1=$windowThick1/$divs

set capLo=$windowHi1
set capMid=$capLo-$capThick/2
set capHi=$capLo-$capThick
set capDiv=$capThick/($divs/2)

set contLo=$capHi
set contHi=$contLo-$contThick
set contDiv=$contThick/($divs/2)

set lightY=$emitterHi1-5

mesh width=$width3d
## X-Mesh
```

```
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv
x.mesh loc=-$capWidth spac=$capWidthDiv
x.mesh loc=$capWidth spac=$capWidthDiv
x.mesh loc=$cellWidthHalf spac=$cellWidthDiv

## Y-Mesh
# Top contact
y.mesh loc=$contHi spac=$contDiv
y.mesh loc=$contLo spac=$contDiv
# Cap
y.mesh loc=$capMid spac=$capDiv
# Window1
y.mesh loc=$windowHi1 spac=$windowDiv1
y.mesh loc=$windowLo1 spac=$windowDiv1
# Emitter1
y.mesh loc=$emitterLo1 spac=$emitterDiv1
# Base1
y.mesh loc=$baseMid1 spac=$baseDiv1
# BSF1
y.mesh loc=$bsfHi1 spac=$bsfDiv1
y.mesh loc=$bsfLo1 spac=$bsfDiv1
# Tunnel12
y.mesh loc=$midTunnel12 spac=0
# Window2
y.mesh loc=$windowHi2 spac=$windowDiv2
y.mesh loc=$windowLo2 spac=$windowDiv2
# Emitter2
y.mesh loc=$emitterLo2 spac=$emitterDiv2
# Base2
y.mesh loc=$baseMid2 spac=$baseDiv2
# BSF2
y.mesh loc=$bsfHi2 spac=$bsfDiv2
y.mesh loc=$bsfLo2 spac=$bsfDiv2
# Tunnel23
y.mesh loc=$midTunnel23 spac=0
# Window3
y.mesh loc=$windowHi3 spac=$windowDiv3
y.mesh loc=$windowLo3 spac=$windowDiv3
# Emitter3
y.mesh loc=$emitterLo3 spac=$emitterDiv3
# Base3
y.mesh loc=$baseLo3 spac=$baseDiv3

## Regions
# Cap
region num=1 material=GaAs x.min=-$capWidth x.max=$capWidth y.min=$capHi
y.max=$capLo
region num=2 material=Vacuum x.min=-$cellWidthHalf x.max=-$capWidth
y.min=$contHi y.max=$capLo
region num=3 material=Vacuum x.min=$capWidth x.max=$cellWidthHalf y.min=$contHi
y.max=$capLo
# Window1 (AlInP)
region num=4 material=InAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi1 y.max=$windowLo1
# Emitter1
region num=5 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$emitterHi1 y.max=$emitterLo1
# Base1
region num=6 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$baseHi1 y.max=$baseLo1
# BSF1 (AlInGaP)
```

```
region num=7 material=InAlAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$bsfHi1 y.max=$bsfLo1
####
# Tunnel1
region num=8 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$bsfLo1 y.max=$windowHi2
####
# Window2
region num=9 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi2 y.max=$windowLo2
# Emitter2
region num=10 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$emitterHi2 y.max=$emitterLo2
# Base2
region num=11 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$baseHi2 y.max=$baseLo2
# BSF2
region num=12 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$bsfHi2 y.max=$bsfLo2
####
# Tunnel2
region num=13 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$bsfLo2 y.max=$windowHi3
####
# Window3
region num=14 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi3 y.max=$windowLo3
# Emitter3
region num=15 material=InGaNAs x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$emitterHi3 y.max=$emitterLo3
# Base3
region num=16 material=InGaNAs x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$baseHi3 y.max=$baseLo3

## Electrodes
electrode name=cathode1 material=Gold x.min=-$capWidth x.max=$capWidth
y.min=$contHi y.max=$contLo
electrode name=anode1 x.min=-$cellWidthHalf x.max=$cellWidthHalf y.min=$bsfLo1
y.max=$bsfLo1
electrode name=cathode2 x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi2 y.max=$windowHi2
electrode name=anode2 x.min=-$cellWidthHalf x.max=$cellWidthHalf y.min=$bsfLo2
y.max=$bsfLo2
electrode name=cathode3 x.min=-$cellWidthHalf x.max=$cellWidthHalf
y.min=$windowHi3 y.max=$windowHi3
electrode name=anode3 x.min=-$cellWidthHalf x.max=$cellWidthHalf y.min=$baseLo3
y.max=$baseLo3

## Doping
# Cap
doping uniform region=1 n.type conc=$capDop
# Window1
doping uniform region=4 n.type conc=$winDop1
# Emitter1
doping uniform region=5 n.type conc=$emitDop1
# Base1
doping uniform region=6 p.type conc=$baseDop1
# BSF1
doping uniform region=7 p.type conc=$bsfDop1
# Window2
doping uniform region=9 n.type conc=$winDop2
# Emitter2
```

```
doping uniform region=10 n.type conc=$emitDop2
# Base2
doping uniform region=11 p.type conc=$baseDop2
# BSF2
doping uniform region=12 p.type conc=$bsfDop2
# Window3
doping uniform region=14 n.type conc=$winDop3
# Emitter3
doping uniform region=15 n.type conc=$emitDop3
# Base3
doping uniform region=16 p.type conc=$baseDop3

## Material properties
# Vacuum (for zero reflection)
material region=2 index.file=Vacuum_AlInP.opt
material region=3 index.file=Vacuum_AlInP.opt
material region=8 index.file=Vacuum_InGaP.opt
material region=13 index.file=Vacuum_InGaP.opt
# GaAs
material material=GaAs EG300=1.424 PERMITTI-VITY=12.9 AFFINITY=4.07 \
  NC300=4.7E17 NV300=9E18 INDEX.FILE=GaAs.opt COPT=7.2E-10 \
  AUGN=5E-30 AUGP=1E-31
# InGaP
material material=InGaP EG300=1.9 PERMITTI-VITY=11.62 AFFINITY=4.16 \
  NC300=1.3E20 NV300=1.28E19 index.file=InGaP.opt COPT=1E-10 \
  MUN=4000 MUP=200 AUGN=3e-30 AUGP=3E-30
# Ge
material material=Ge EG300=0.661 PERMITTI-VITY=16.2 AFFINITY=4 \
  NC300=1E19 NV300=5E18 index.file=Ge.opt COPT=6.41E-14 \
  MUN=3900 MUP=1900 AUGN=1E-30 AUGP=1E-30
# AlGaAs
material material=AlGaAs MUN=9000 MUP=100 INDEX.FILE=AlGaAs.opt
# AlInP (=InAsP)
material material=InAsP EG300=2.4 PERMITTI-VITY=11.7 AFFINITY=4.2 \
  NC300=1.08E20 NV300=1.28E19 index.file=AlInP.opt COPT=1.2E-10 \
  MUN=2291 MUP=142 AUGN=9E-31 AUGP=9E-31
# AlInGaP (=InAlAsP)
material material=InAlAsP EG300=2.4 PERMITTI-VITY=11.7 AFFINITY=4.2 \
  NC300=1.2E20 NV300=1.28E19 index.file=AlInP.opt COPT=1E-10 \
  MUN=2150 MUP=141 AUGN=3e-30 AUGP=3E-30
# InGaNAs
material material=InGaNAs EG300=1.0 PERMITTI-VITY=11.7 AFFINITY=4.05 \
  NC300=3.2e19 NV300=1.8e19 index.file=InGaNAs.opt COPT=7.2e-10 \
  MUN=3000 MUP=150
# Gold
material material=Gold real.index=1.2 imag.index=1.8

## Models
### here temperature is set for the run ###
models region=1 CONMOB temp=300 print
models region=10 CONMOB temp=300 print
models region=11 CONMOB temp=300 print
models OPTR BGN temp= 300 print


## Light beams
#b1,630 b2,825
beam num=1 x.origin=0 y.origin=$lightY angle=90 back.refl
power.file=AM0nrel.spec \
  wavel.start=0.12 wavel.end=0.75 wavel.num=630
beam num=2 x.origin=0 y.origin=$lightY angle=90 back.refl
power.file=AM0nrel.spec \
```

```
  wavel.start=0.7501 wavel.end=2.4 wavel.num=825

### as with the single junction cell, if a structure file is needed for plotting
###
#struct outfile=InGaP_GaAs_Ge.str
#tonyplot InGaP_GaAs_Ge.str

solve init
method gummel newton maxtraps=10 itlimit=25
solve b1=0.9 b2=0.9

method newton maxtraps=10 itlimit=100
solve b1=0.95 b2=0.95
extract name="isc1" max(i."cathode1")
set isc1=$isc1*$width3d
set i0a=$isc1-0.00001
set i1a=$i0a-0.0001
set i2a=$i1a-0.0001
set i3a=$i2a-0.0001
set i4a=$i3a-0.0001
set i5a=$i4a-$isc1/80
set i6a=$i5a-$isc1/80
set i7a=$i6a-$isc1/40
set i8a=$i7a-$isc1/40
set i9a=$i8a-$isc1/40
set i10a=$i9a-$isc1/40
set i11a=$i10a-$isc1/20

method newton maxtraps=10 itlimit=100
solve b1=0.95 b2=0.95
extract name="isc2" max(i."cathode2")
set isc2=$isc2*$width3d
set i0b=$isc2-0.00001
set i1b=$i0b-0.0001
set i2b=$i1b-0.0001
set i3b=$i2b-0.0001
set i4b=$i3b-0.0001
set i5b=$i4b-$isc2/80
set i6b=$i5b-$isc2/80
set i7b=$i6b-$isc2/40
set i8b=$i7b-$isc2/40
set i9b=$i8b-$isc2/40
set i10b=$i9b-$isc2/40
set i11b=$i10b-$isc2/20

method newton maxtraps=10 itlimit=100
solve b1=0.95 b2=0.95
extract name="isc3" max(i."cathode3")
set isc3=$isc3*$width3d
set i0c=$isc3-0.00001
set i1c=$i0c-0.0001
set i2c=$i1c-0.0001
set i3c=$i2c-0.0001
set i4c=$i3c-0.0001
set i5c=$i4c-$isc3/80
set i6c=$i5c-$isc3/80
set i7c=$i6c-$isc3/40
set i8c=$i7c-$isc3/40
set i9c=$i8c-$isc3/40
set i10c=$i9c-$isc3/40
set i11c=$i10c-$isc3/20
```

```
log outfile=InGaP_GaAs_Ge.log

method newton maxtraps=10 itlimit=100
solve b1=0.95 b2=0.95

method newton maxtraps=10 itlimit=100
contact num=2 current
## Pmax points [InGaP 18-25]
solve i2=-$i0a b1=0.95 b2=0.95
solve i2=-$i1a b1=0.95 b2=0.95
solve i2=-$i2a b1=0.95 b2=0.95
solve i2=-$i3a b1=0.95 b2=0.95
solve i2=-$i4a b1=0.95 b2=0.95
solve i2=-$i5a b1=0.95 b2=0.95
solve i2=-$i6a b1=0.95 b2=0.95
solve i2=-$i7a b1=0.95 b2=0.95
solve i2=-$i8a b1=0.95 b2=0.95
solve i2=-$i9a b1=0.95 b2=0.95
solve i2=-$i10a b1=0.95 b2=0.95
solve i2=-$i11a b1=0.95 b2=0.95
solve i2=0 b1=0.95 b2=0.95

contact num=4 current
## Pmax points [GaAs 15-25]
solve i4=-$i0b b1=0.95 b2=0.95
solve i4=-$i1b b1=0.95 b2=0.95
solve i4=-$i2b b1=0.95 b2=0.95
solve i4=-$i3b b1=0.95 b2=0.95
solve i4=-$i4b b1=0.95 b2=0.95
solve i4=-$i5b b1=0.95 b2=0.95
solve i4=-$i6b b1=0.95 b2=0.95
solve i4=-$i7b b1=0.95 b2=0.95
solve i4=-$i8b b1=0.95 b2=0.95
solve i4=-$i9b b1=0.95 b2=0.95
solve i4=-$i10b b1=0.95 b2=0.95
solve i4=-$i11b b1=0.95 b2=0.95
solve i4=0 b1=0.95 b2=0.95

contact num=6 current
## Pmax points [InGaNAs 16-25]
solve i6=-$i0c b1=0.95 b2=0.95
solve i6=-$i1c b1=0.95 b2=0.95
solve i6=-$i2c b1=0.95 b2=0.95
solve i6=-$i3c b1=0.95 b2=0.95
solve i6=-$i4c b1=0.95 b2=0.95
solve i6=-$i5c b1=0.95 b2=0.95
solve i6=-$i6c b1=0.95 b2=0.95
solve i6=-$i7c b1=0.95 b2=0.95
solve i6=-$i8c b1=0.95 b2=0.95
solve i6=-$i9c b1=0.95 b2=0.95
solve i6=-$i10c b1=0.95 b2=0.95
solve i6=-$i11c b1=0.95 b2=0.95
solve i6=0 b1=0.95 b2=0.95

log off
```

APPENDIX B. MATLAB SOURCE CODE

A. FILERW.M

For a single change in one parameter, use filerw.m function

```
function filerw(file,old,new)
%This program opens an infile, "file", and writes over the
%"old" anode thickness with the "new" anodethickness
%file must be in ''
%old and new are the old and new values of anodethickness
%This same function was also used to iterate other variables by replac-
ing
%‘anodethickness’ with the name of the variable to be iterated.
%
%From B.P. Davenport [28]
fidr=fopen(file,'r');
a=fscanf(fidr,'%c');
fclose(fidr);
fidw=fopen(file,'w');
a=strrep(a,sprintf('baseThick1=%g',old),sprintf('baseThick1=%g',new));
fwrite(fidw,a);

fclose(fidw);
```

For a multi-variable change, modify filerw3j with the names of the parameters to be changed within the DeckBuild input file.

```
function filerwtj3(file,old,old2,new,new2)
%This program opens an infile, "file", and writes over the
%"old" anode thickness with the "new" anodethickness
%file must be in ''
%old and new are the old and new values of anodethickness
%This same function was also used to iterate other variables by replac-
ing
%‘anodethickness’ with the name of the variable to be iterated.
%
%modified from B.P. Davenport [28]
fidr=fopen(file,'r');
a=fscanf(fidr,'%c');
fclose(fidr);
fidw=fopen(file,'w');

a=strrep(a,sprintf('emitDop1temp=%g',old),...
sprintf('emitDop1temp=%g',new));

a=strrep(a,sprintf('emitDop2temp=%g',old2),...
sprintf('emitDop2temp=%g',new2));
fwrite(fidw,a);
fclose(fidw);
```

B. MJ_IVMAXP.M

From Bates, a function to calculate and plot the *I-V* characteristics of a simulation [3]

```
function [isctot,voctot,imptot,vmptot,pmaxtot,Eff]=mj_ivmaxp...
(infile)
%MJ_I-VMAXP Extract and plot solar cell properties from ATLAS log file.
%
% [isctot,voctot,imptot,vmptot,pmaxtot,fftot]=mj_ivmaxp(runinfile)
% 'infile' is ATLAS log filename (with or without '.log')
% 'isctot' is short-circuit current (A)
% 'voctot' is open-circuit voltage (V)
% 'imptot' is maximum-power current (A)
% 'vmptot' is maximum-power voltage (V)
% 'pmaxtot' is maximum output power (W)
% 'fftot' is fill factor (W/W)
% 'Eff' is efficiency (%) at 1353W/m^2
%
% IMPORTANT: Correct units for output values anticipates scaling
% of ATLAS simulation to give output in correct units
%
% IMPORTANT: Assumes current values in ATLAS log file are
% monotonically decreasing (from Isc to 0)
format long;
% determines if infile has '.log' or not
if infile(length(infile)-3:length(infile))=='.log'
runinfile=infile(1:length(infile)-4);
end
% reads in basic parameters from infile
datacol=textread( [runinfile '.log'],'%*s%u%* [^\n]','headerlines',18);
numelect=datacol(1);
cols=datacol(2);
% determines number of beams used and adjusts appropriately
beams=mod(cols-4,numelect*3)+1;
beamstuff= [];
for i=1:beams
beamstuff= [beamstuff '%*f'];
end
% following textread string is for n-on-p, use commented statement
% if cell utilizes p-on-n solar cells
trodestuff= ['%*f%*f%f%f%*f%*f']; % n-on-p
% trodestuff= ['%f%*f%*f%*f%*f%f']; % p-on-n
% uses textread to extract data from ATLAS log file
pwredge=0; badpmax=0;
for i=1:(numelect/2)
 [Io(:,i) Vo(:,i)]=textread( [runinfile '.log'], ['%*s' beamstuff ...
'%*f%*f%*f' trodestuff '%* [^\n]'],'headerlines',20);
trodestuff= ['%*f%*f%*f%*f%*f%*f' trodestuff];
Po(:,i)=Io(:,i).*Vo(:,i);
isc(i)=max(Io(:,i));
 [mincurrent indx]=min(Io(:,i));
voc(i)=Vo(indx,i);
 [Pmax(i) indx]=max(Po(:,i));
```

```
% deals with spurious convergences
while Vo(indx,i)>Vo(indx+1,i)
disp( ['*** SUSPICIOUS PMAX' num2str(i) '=' num2str(Pmax(i)) ...
' DROPPED ***']);
 [Pmax(i) addon]=max(Po((indx+1):max(size(Po(:,i))),i));
indx=indx+addon;
badpmax=1;
end
% monitors for bounding of pmax point
if indx==2
pwredge=1;
disp( ['*** INCOMPLETE LOWER BOUNDING OF PMAX' num2str(i) ....
' ***']);
numboundprob=numboundprob+1;
elseif indx==(max(size(Po(:,i)))-1)
pwredge=2;
disp( ['*** INCOMPLETE UPPER BOUNDING OF PMAX' num2str(i) ....
' ***']);
numboundprob=numboundprob+1;
end
FF(i)=Pmax(i)/(isc(i)*voc(i));
imp(i)=Io(indx,i);
vmp(i)=Vo(indx,i);
end
% calls to maxpower.m to search for multi-junction total power
 [pmaxtot,imptot,itotal,vtotal]=maxpower(Io,Vo,imp,isc,voc,numelect);
% assigns outputs and plots
isctot=max(itotal);
voctot=max(vtotal);
vmptot=pmaxtot/imptot;
fftot=pmaxtot/(isctot*voctot);
%%
Eff=100*pmaxtot/(.1353);
%%
Vtotmax=vtotal;
Iomax=Io;
Itotmax=itotal;
xlim=1.1*max(vtotal);
ylim=1.1*max(isc);
textposit1=1.05;
textposit2='bottom';
% differing plot functions for differing number of layers
if (numelect/2)==4
plot(Vo(:,1),Io(:,1),'b',Vo(:,2),Io(:,2),'r',Vo(:,3),Io(:,3),...
'g',Vo(:,4),Io(:,4),'k','LineWidth',2);
hold on;
plot(vtotal,itotal,'m','LineWidth',3);
legend('InGap','GaAs','InGaNAs','Ge','Total Cell',0);
elseif (numelect/2)==3
plot(Vo(:,1),Io(:,1),'b',Vo(:,2),Io(:,2),'r',...
Vo(:,3),Io(:,3),'g','LineWidth',2);
hold on;
plot(vtotal,itotal,'m','LineWidth',3);
legend('InGap','GaAs','Ge','Total Cell',0);
elseif (numelect/2)==1
```

```
plot(Vo(:,1),Io(:,1),'b','LineWidth',2);
hold on;
textposit1=0.95
textposit2='top';
end
plot(vmptot,imptot,'o','LineWidth',2,'MarkerEdgeColor','k',...
'MarkerFaceColor','r');
hold off;
text(vmptot,textposit1*isctot, ['P_{max} = ' num2str(pmaxtot*1000)...
' mW/cm^{2}'],'VerticalAlignment',textposit2,...
'HorizontalAlignment','right','FontSize',14);
% title( ['Data From ' runinfile '.log']);
xlabel('Voltage (V)');
ylabel('Current (A)');
axis( [0 xlim 0 ylim]);
```

C. MAXPOWER.M

Used in `mj_ivmaxp.m` to get *I-V* characteristics [3].

```
function [maxp,imaxp,itotal,vtotal]=maxpower(Io,Vo,imp,isc,voc,numelect)
% creates a vector of currents to solve for maximum overall power
% first line gets 10 currents between min Imp and min Isc
itry=linspace(min(imp),min(isc),10);
% second line adds 10 currents outside bounds to make a better I-V curve
itry= [0 linspace(min(imp)*0.6,min(imp),10) itry];
%% parameters for interpolation
% Io=known y's (decreasing)
% Vo=known x's (increasing)
% itry=given y's
% vtgt=target x's
% this 'for' loop determines the start and end of each junction layer's
% current and voltage information in the ATLAS log file
istart(1)=2;
for i=1:(numelect/2)-1
for j=istart(i):max(size(Io(:,i)))
if Io(j,i)<0.00001
istart(i+1)=j+1;
break;
end
end
end
istart((numelect/2)+1)=max(size(Io(:,1)))+1;
% this 'for' loop performs the interpolation for the voltage from
% each junction layer at the currents in 'itry' and adds them
for j=1:max(size(itry))
maxpwr(j)=0;
vtotal(j)=0;
for i=1:(numelect/2)
pivot=0;
for x=istart(i):(istart(i+1)-1)
if Io(x,i)<itry(j)
```

```
pivot=x;
if pivot==istart(i)
pivot=istart(i)+1;
end
end
if pivot
break;
end
end
if ~pivot
pivot=istart(i+1)-1;
end
linterp=(Io(pivot,i)-itry(j))/(Io(pivot,i)-Io(pivot-1,i));
vtgt=Vo(pivot,i)-((Vo(pivot,i)-Vo(pivot-1,i))*linterp);
vtotal(j)=vtotal(j)+vtgt;
maxpwr(j)=maxpwr(j)+(itry(j)*vtgt);
end
end
% these vectors represent the total I-V curve for the MJ cell
itotal= [itry min(isc)];
vtotal= [vtotal 0];
% maximum power for full cell is highest power achieved over all
% currents in 'itry'
 [maxp indx]=max(maxpwr);
imaxp=itry(indx);
```

D. EFF_PMAX_PLOT.M

Used in `inter_test_XX.m` to plot 2D and 3D changes in pmax and efficiency.

For 2D:

```
function eff_pmax_2Dplot(data, iterations)
figure(10)
plot(data(1:1+iterations,2),data(1:1+iterations,8))
hold
xlabel('parameter')
grid on
title ('Varying Eff')
plot(data(1+(iterations+1):2*(1+iterations),2),data(1+(iterations+1):2*(
1+iterations),8),'g')
plot(data(2*(iterations+1)+1:3*(1+iterations),2),data(2*(iterations+1)+1
:3*(1+iterations),8),'r')
plot(data(3*(iterations+1)+1:4*(1+iterations),2),data(3*(iterations+1)+1
:4*(1+iterations),8),'c')
plot(data(4*(iterations+1)+1:5*(1+iterations),2),data(4*(iterations+1)+1
:5*(1+iterations),8),'m')
plot(data(5*(iterations+1)+1:6*(1+iterations),2),data(5*(iterations+1)+1
:6*(1+iterations),8),'y')
```

```
figure(11)
plot(data(1:1+iterations,2),data(1:1+iterations,7))
hold
xlabel('parameter')
grid on
title ('Varying Pmax')
plot(data(1+(iterations+1):2*(1+iterations),2),data(1+(iterations+1):2*(
1+iterations),7),'g')
plot(data(2*(iterations+1)+1:3*(1+iterations),2),data(2*(iterations+1)+1
:3*(1+iterations),7),'r')
plot(data(3*(iterations+1)+1:4*(1+iterations),2),data(3*(iterations+1)+1
:4*(1+iterations),7),'c')
plot(data(4*(iterations+1)+1:5*(1+iterations),2),data(4*(iterations+1)+1
:5*(1+iterations),7),'m')
plot(data(5*(iterations+1)+1:6*(1+iterations),2),data(5*(iterations+1)+1
:6*(1+iterations),7),'y')
```

For 3D:

```
function eff_pmax_plot(data, iterations)

figure(45)
plot3(data(1:1+iterations,1),data(1:1+iterations,2),data(1:1+iterations,
8))
hold
xlabel('first parameter')
ylabel('second parameter')
zlabel('%')
grid on
title ('Varying Eff')
plot3(data(1+(iterations+1):2*(1+iterations),1),data(1+(iterations+1):2*
(1+iterations),2),data(1+(iterations+1):2*(1+iterations),8),'g')
plot3(data(2*(iterations+1)+1:3*(1+iterations),1),data(2*(iterations+1)+
1:3*(1+iterations),2),data(2*(iterations+1)+1:3*(1+iterations),8),'r')
plot3(data(3*(iterations+1)+1:4*(1+iterations),1),data(3*(iterations+1)+
1:4*(1+iterations),2),data(3*(iterations+1)+1:4*(1+iterations),8),'c')
plot3(data(4*(iterations+1)+1:5*(1+iterations),1),data(4*(iterations+1)+
1:5*(1+iterations),2),data(4*(iterations+1)+1:5*(1+iterations),8),'m')
plot3(data(5*(iterations+1)+1:6*(1+iterations),1),data(5*(iterations+1)+
1:6*(1+iterations),2),data(5*(iterations+1)+1:6*(1+iterations),8),'y')

hold

figure(46)
plot3(data(1:1+iterations,1),data(1:1+iterations,2),data(1:1+iterations,
7))
hold
xlabel('first parameter')
ylabel('second parameter')
zlabel('W')
grid on
title ('Varying Pmax')
plot3(data(1+(iterations+1):2*(1+iterations),1),data(1+(iterations+1):2*
(1+iterations),2),data(1+(iterations+1):2*(1+iterations),7),'g')
```

```
plot3(data(2*(iterations+1)+1:3*(1+iterations),1),data(2*(iterations+1)+
1:3*(1+iterations),2),data(2*(iterations+1)+1:3*(1+iterations),7),'r')
plot3(data(3*(iterations+1)+1:4*(1+iterations),1),data(3*(iterations+1)+
1:4*(1+iterations),2),data(3*(iterations+1)+1:4*(1+iterations),7),'c')
plot3(data(4*(iterations+1)+1:5*(1+iterations),1),data(4*(iterations+1)+
1:5*(1+iterations),2),data(4*(iterations+1)+1:5*(1+iterations),7),'m')
plot3(data(5*(iterations+1)+1:6*(1+iterations),1),data(5*(iterations+1)+
1:6*(1+iterations),2),data(5*(iterations+1)+1:6*(1+iterations),7),'y')
hold
```

E. TIME.M

Used as a diagnostic tool to determine the amount of time for each run.

```
function [T]=time(s)
%This function takes an input time, 's',
%in seconds and returns time in the format
%of hr:min:sec
%
%From B.P. Davenport [28]
hr=floor(s/3600);
min=floor(s/60-hr*60);
sec=floor(s-min*60-hr*3600);
if(hr<10)
hr=sprintf('0%u',hr);
else
hr=sprintf('%u',hr);
end
if(min<10)
min=sprintf('0%u',min);
else
min=sprintf('%u',min);
end
if(sec<10)
sec=sprintf('0%u',sec);
else
sec=sprintf('%u',sec);
end
T=sprintf('%s:%s:%s',hr,min,sec);
```

F. INTER_TEST_XX.M

Multiple variations of Davenport's `ATLASRUN.m` were used in this thesis [28]. A single run m-file was developed to permit a plot and data to be generated for a single change. In the iterative testing, inner and outer loops were run to vary two parameters at a time, similar to Canfield's `ATLASRUN_PARAM1_PARAM2.m` [7]

1. Single Run Testing File

```
% inter_test_mj_singlerun.m
% a modified version of Davenport's file [28].
close all
clear all
clc


k=0;
totalruntime=0;
tic
k=k+1;
!C:\Silvaco\etc\GuiAppStarter.exe -lib-dir-name deckbuild -exe-name
Deckbld -run InGaP_GaAs_Ge.in -outfile atlaslog.log
sprintf('Executing run %u\nStandby for results',k)
xy=-1;
while(xy==-1)
xy=fopen('done.log');
end


[isctot,voctot,imptot,vmptot,pmaxtot,Eff]=mj_ivmaxp('InGaP_GaAs_Ge.log')
data(k,:)= [isctot voctot imptot vmptot pmaxtot Eff]

clc
data
thisruntime=toc;T1=time(toc);
totalruntime=totalruntime+thisruntime;T2=time(totalruntime);
averageruntime=totalruntime/k;T3=time(averageruntime);
fclose('all')

save('doping_best_3_1_375_b1at1pt5', 'data')
```

2. Iterative Testing File

```
%inter_test_mj.m
%based on Canfield's work [7].
close all
clear all
clc

iterations=5;
old=0.26;ol=old;
 old2=1.3;ol2=old2;

initval=old;
initvalb=old2;
final=.4;
final2=2.6;
step=(final-old)/iterations;
step2=(final2-old2)/iterations;
```

```
k=0;
totalruntime=0;

for(a=ol:step:final)
     for(b=ol2:step2:final2)
tic
k=k+1;
fidres=fopen('atres.txt','a')
filerwtj3('InGaP_GaAs_Ge_sm.in',old,old2,a, b);
%calls DeckBuild input deck to run ATLAS
!C:\Silvaco\etc\GuiAppStarter.exe -lib-dir-name deckbuild -exe-name
Deckbld -run InGaP_GaAs_Ge_sm.in -outfile atlaslog.log
sprintf('Executing run %u\nStandby for results',k)
xy=-1;
while(xy==-1)
xy=fopen('done.log');
end
%solves for I-V characteristics and plot I-V curve

[isctot,voctot,imptot,vmptot,pmaxtot,Eff]=mj_ivmaxp('InGaP_GaAs_Ge_sm.lo
g')
data(k,:)= [a b isctot voctot imptot vmptot pmaxtot Eff]
fprintf(fidres,'%f %f %f %f %f %f\n',data(k,:));

clc
data
%use to keep track of time
thisruntime=toc;T1=time(toc);
totalruntime=totalruntime+thisruntime;T2=time(totalruntime);
averageruntime=totalruntime/k;T3=time(averageruntime);
estimateruntime=averageruntime*(iterations+1);T4=time(estimateruntime);
sprintf('This run took %s\nTotal run time so far is %s\nAverage run time
is %s\nEstimated total run time is %s',T1,T2,T3,T4)
%sets values for next iteration
old=a;
old2=b;
fclose('all')
    end
%resets the alter parameter to the original value to start next
iteration
filerwtj3('InGaP_GaAs_Ge_sm.in',old,old2,a, initvalb);
end
%saves the data to a mat-file for future recall
save('base_1-2at400K_small', 'data')
%plots data
eff_pmax_plot(data, iterations);
eff_pmax_2Dplot(data, iterations);

%resets DeckBuild file to original values
filerwtj3('InGaP_GaAs_Ge_sm.in',old,old2,initval, initvalb);
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] *Solar Cell* webpage (http://en.wikipedia.org/wiki/Solar_cell), last accessed 7 August 2007.

[2] Hojun Yoon, “Multijunction space solar cells with 30% efficiency and beyond,” presented at the Aerospace Corporation Space Power Workshop, Los Angeles, CA, 2007.

[3] Andrew D. Bates, “Novel optimization techniques for multijunction solar cell design using Silvaco ATLAS,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 2004.

[4] Darin J. McCloy, “High efficiency solar cells: a model in Silvaco,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 1999.

[5] Panayiotis Michalopoulos, “A novel approach for the development and optimization of state-of-the-art photovoltaic devices using Silvaco,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 2002.

[6] Max Green, “The verification of Silvaco as a solar cell simulation tool and the design and optimization of a four-junction solar cell,” M.S. thesis, Naval Postgraduate School, 2002.

[7] Burt Canfield, “Advanced modeling of high temperature performance of indium gallium arsenide thermophotovoltaic cells,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 2005.

[8] E. B. Linder and J. P. Hanley, “Manufacturing experience with GaInP2/GaAs/Ge solar panels for space demonstration,” in *Proceedings of the 25th Photovoltaic Specialists Conference*, 1996, pp. 267-270.

[9] Ben G. Streetman and Sanjay Banerjee, *Solid State Electronic Devices*. New Jersey: Princeton Hall, 2000.

[10] “Introduction to the Space Environment,” class notes for PH 2514, Department of Physics, Naval Postgraduate School, Monterey, CA, Winter 2006.

[11] *Corrosion source* webpage (http://www.corrosionsource.com/handbook/periodic/periodic_table.gif), last accessed 16 August 2007.

[12] Robert F. Pierret, *Semiconductor Device Fundamentals*, New York: Addison-Wesley, 1996.

[13] “Space Power and Radiation Effects,” class notes for EC3230, Department of Electrical and Computer Engineering, Naval Postgraduate School, Spring 2007.

[14] S.O. Kasap, *Principles of Electronic Materials and Devices*, McGraw Hill, New York, 2002.

[15] *Solar Spectra: Air Mass Zero* webpage (http://rredc.nrel.gov/solar/spectra/am0/E490_00a_AM0.xls), last accessed 17 August 2007.

[16] *Spectrolabs Photovoltaic Products 28.3% Ultra Triple Junction (UTJ) Solar Cells* webpage, (http://www.spectrolab.com/DataSheets/TNJCell/utj3.pdf), last accessed 17 August 2007.

[17] Baldomero Garcia, “Indium gallium nitride multijunction solar cell simulation using Silvaco ATLAS,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 2007.

[18] Geoffrey A. Landis, Danielle Merritt, Ryne P. Raffaelle, and David Scheiman, “High temperature solar cell development,” in NASA *18th Space Photovoltaic Research and Technology Conference*, 2005, pp. 108-115.

[19] Martin A. Green, “General temperature dependence of solar cell performance and implications for device modeling,” in *Progress in Photovoltaics: Research and Applications*, 2003, pp. 333-340.

[20] S. M. Sze, Physics of Semiconductor Devices, John Wiley & Sons: New York, 1981.

[21] Zhihong Lin, Xiaokang Huang, and W. B. Berry, “Temperature dependence on the energy gap selection in two cell, four terminal tandem solar cell design,” in *Proceedings of the 20th Photovoltaic Specialists Conference,* 1991, pp. 323-328.

[22] *ATLAS Users Manual: Device Simulation Software*, SILVACO International, 2004.

[23] *Solar Cells* webpage, (http://www.corrosion-doctors.org/Solar/cells.htm), last accessed 18 August 2007.

[24] *Silvaco Virtual Wafer Fab specifications* webpage, (http://www.silvaco.com/products/interactive_tools/vwf.pdf), last accessed 18 August 2007.

[25] R. Jones (private communication), 2007.

[26] Otfried Madelung, *Semiconductors: Data Handbook*, Germany: Springer, 2003.

[27] D. A. Allwood, P. C. Klipstein, N. J. Mason, R. J. Nicholas, and P. J. Walker, “GaAs high temperature optical constants and application to optical monitoring within the MOVPE environment,” in *Journal of Electronic Materials*, 2000, pp. 99-105.

[28] Bradley P. Davenport, “Advanced thermophotovoltaic cells modeling, optimized for use in radioisotope thermoelectric generators (RTGS) for mars and deep space missions,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 2004.

[29] Robert Gelinas, “A novel approach to modeling tunnel junction diodes using Silvaco ATLAS,” M.S. thesis, Naval Postgraduate School, Monterey, CA, 2005.

[30] P. K. Chiang, J. H. Ermer, W. T. Nishikawa, D. D. Krut, D. E. Joslin, J. W. Eldredge, and B. T. Cavicchi, “Experimental results of GaInP2/GaAs/Ge triple junction cell development for space power systems,” in *Proceedings of the 25th Photovoltaic Specialists Conference*, 1996, pp. 183-186.

[31] T. Redick (private communication), 2007.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Sherif Michael
   Naval Postgraduate School
   Monterey, California

4. Todd Weatherford
   Naval Postgraduate School
   Monterey, California

5. Michael Sanders
   Naval Postgraduate School
   Monterey, California